Spring 5-2008

Robotics Senior Capstone Interim Report The Mobile Human Seeking Robot

Brent Randall Beatty
University of Tennessee - Knoxville

Gregory William Smolen
University of Tennessee- Knoxville

THE UNIVERSITY OF TENNESSEE, KNOXVILLE

Robotics Senior Capstone Interim Report

The Mobile Human Seeking Robot

Brent Beatty

Gregory Smolen

Design Team

Jason Alfrey
Brent Beatty
Claire Chisholm
Scott Frazier
Hoa Hoang
Greg Smolen

Submitted to Lead Professor:

Dr. William R. Hamel

Submitted on:

December 13, 2007

Table of Contents

1. OVERVIEW 7

1.1 Introduction 7

1.2 Terms, Definitions, and Acronyms 7

1.3 Background and Applicability 7

1.4. REQUIREMENTS 8

1.4.1. Operational Environment 8

1.4.2. Scenario 9

1.4.3. Functional Requirements 10

1.4.4. Proposed System 10

1.4.5. Sequence of Operations 10

1.4.6 MoHSeR Benefits 11

1.4.7 Capabilities Required 11

2. SYSTEM ARCHITECTURE 12

3. SENSOR SUITE 14

3.1. PYROELECTRIC 14

3.1.1 Pyroelectricity 14

3.1.2. Fresnel Lens 17

3.1.3 Sensor Solution 18

3.1.4. Sensor/Robot Interface 19

3.1.5. References 20

3.2 MICROWAVE MOTION DETECTOR 20

3.2.1 Overview/Method of Operation/Technical Status 20

3.2.2 Hardware Requirements 21

3.2.3 Input/ Output 22

3.2.4 Software Requirements 22

3.3 VIDEO 23

3.3.1 Overview 23

3.3.2 Technical Status and Considerations 23
3.3.3 Hardware Requirements 24
3.3.3.1 Camera 24
3.3.3.2 Wireless Receiver 25
3.3.3.3 Analog to Digital Converter 26
3.3.3.4 Operational Modes/Software Requirements 26
3.3.4 Specifications/Documentation 26
3.4 OBSTACLE DETECTION 27
3.4.1 Scope 27
3.4.2 Technical Status and Considerations 27
3.4.3 Obstacle Detection Solution 27
3.4.3.1 LV-MaxSonar-EZ1Sensor 28
3.4.3.2 OOPic 29
3.5 TRANSMITTER 30
3.5.1 SCOPE 30
3.5.2 BLUETOOTH 30
3.5.3 TECHNICAL STATUS 31
4. PLATFORMS 31
4.1 CREATE 31
4.1.1 Introduction 31
4.1.2 Navigation 32
4.1.2.1 Scope 32
4.1.2.2 Technical Status 36
4.1.3 Interface 36
4.1.4 Adaptive Mapping and Path Planning 37
4.2 DARPA 38
4.2.1 Introduction 38
4.2.2 Current Status 39
5. INTEGRATION 41
5.1 Introduction 41

5.1.1 System Diagram 41
5.2.1 OOPic 42
5.2.2 Pan System 43
5.3 Material Requirements 43
5.4 SolidWorks 44
6. Experimentation 47
6.1 Overview 47
6.2 Component Testing: Pyroelectric 47
6.3 Component Testing: Microwave Motion Detector 54
6.4 Component Testing: Sensors (Odometer, Beacons, Transmitter, Compass, Sonar) 55
6.5 Sensors – Camera 57
6.6 Platforms Test: iRobot Create 58
6.7 Integration 59
6.9 Final Demonstration 60
7. PROJECT STATUS 62
7.1 Executive Summary 62
7.1 Progress Report 62
7.2 Project Timeline 64
7.3 Budget 65
7.4 Work Break-Down Structure 66
7.5 Project Management Techniques 66
8. CONCLUSION 69
9. APPENDIX 70

1. OVERVIEW

1.1 Introduction

It is the intent of the senior capstone program to allow a venue for senior engineering students to exercise their knowledge and application of the engineering design process from inception to completion under the direction of an experienced and professional engineer. In doing so, the student team identifies a technological challenge and develops a solution. The purpose of this report is to convey the efforts the robotics senior capstone design team has accomplished in the first phases of the engineering process.

1.2 Terms, Definitions, and Acronyms

MoHSeR -Mobile Human Seeking Robot
PIR – Pyroelectric Infrared
SLAM – Simultaneous Location and Mapping
DARPA – Defense Advanced Research Projects Agency
Pyroelectric – A material that is capable of changing thermal energy to electrical energy
FORs – Functional and Operational Requirements
I2C – Inter-Integrated Circuit
PWM – Pulse Width Modulation
BAM – Bluetooth Adapter Module
PIC – Programmable Intelligent Computer
RRV – Robotics Research Vehicle

1.3 Background and Applicability

Mobile robotic platforms and autonomous sensors have proven to be highly effective in their respective applications, including manufacturing and military applications, but integrating these systems together and applying them into civilian society has been less than common. Yet, several companies are now poised to deliver such systems along with claims of cost savings, increased safety, and better efficiency. Research is being conducted accordingly[1]. It is our goal to replicate such a search and rescue technology device. These devices are capable of replacing human search teams inside potentially hazardous or dangerous environments. They can also be deployed in applications which require more scrutiny than can be provided by employment of human security personnel. Current search options put human personnel at risk in order to confirm the presence of persons trapped inside or intruding these hazardous environments. It is necessary to develop a means to confirm the presence of and verify the location of human victims with minimal endangerment to additional human life.

[1] AUVSI International Aerial Robotics Competition hosts an annual competition for non-aerial remote surveillance of building interiors. Mobile Robots Inc markets the cost savings of using mobile surveillance.

The University of Tennessee robotics senior capstone design team examined the need and application of sensors on mobile robotic platforms. More specifically, the team researched the application of human identification sensors, most specifically pyroelectric sensors. This was directly related to the perceived need for the development of robotic devices for remote surveillance applications. It was desired to apply a pyroelectric sensor in a non-standard application. Ergo, with these initial parameters, a scenario was developed to test these specific conditions. The overall scenario became the implementation of a sensor suite, or a system of sensors, that would operate together in order to relay information wirelessly to a user. The information of interest is the presence and location of a human.

Although the principle of the sensor suite is to stand alone in functionality and be applied to any mobile platform with minimal changes, the Create mobile platform is the primary system used for this project. As resources allow, MoHSeR plans to continue the project by applying the sensor suite to the RRV.

1.4. REQUIREMENTS

1.4.1. Operational Environment

It is proposed that MoHSeR be deployed in the fourth floor environment of the Dougherty Engineering building in after hours conditions. Figure 1.1 displays the envisioned operational environment. MoHSeR will be limited to performing its tasks to indoor locations. This is representative of a nuclear storage facility, secured office complex, or similar situations that are restricted to essential access only. The operational environment includes standard professional flooring finishes; MoHSeR must operate in varied indoor terrains whether tile, cement, or carpet. Notice Figure 1.1 only outlines the environment in which MoHSeR will operate; it is desired that MoHSeR determine the obstacles within each room using a form of SLAM. The fourth floor of the Dougherty Engineering building will provide a setting to demonstrate functional operation.

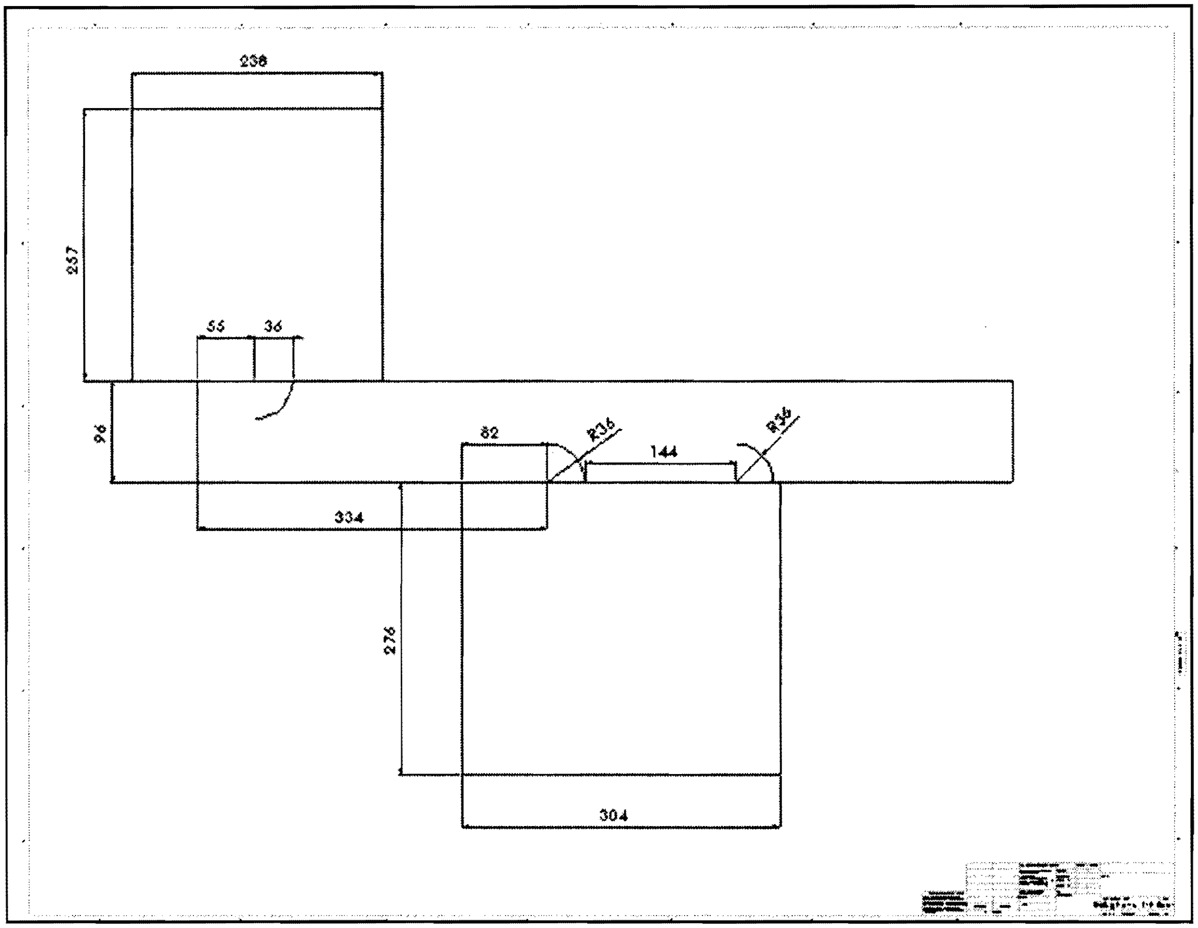


Figure 1.1: Map of the Operational Environment without Obstacles

1.4.2. Scenario

The proposed scenario is to deploy MoHSeR remotely in a hazardous environment building situation. This remote building will have known paths. The facility will have low light conditions, and the environment will be equipped with conventional industrial heating and cooling systems. The robot will be deployed on a defined path with the robot not expecting any movement or unexpected obstacles in its surroundings. There will be no changes in the elevation, yet MoHSeR must be able to accept changes in flooring type. MoHSeR assumes that any intruders will be clothed, on the floor, walking or standing, and have a heat signature. MoHSeR must be minimally intrusive to its environment.

Since it is operating in low light conditions, MoHSeR will be continuously seeking heat signatures. Once a heat signature is detected a low-light camera will be used to record the environment and a motion detector will be sampled to determine if the heat signature is in motion. This information will be relayed wirelessly to a human operator, or controller, no more than 325 ft away from the MoHSeR. The information relayed to the operator will provide a three-component message with heat signature information, visual confirmation, and whether or not the heat signature is in motion. This will allow the operator to discern what further action is necessary.

1.4.3. Functional Requirements

In order to address the proposed scenario appropriately there must be a clear set of functions MoHSeR must be able to accomplish, these are as follows:

- Any heat signature in the environment must be detected
- Must be able to identify a human on the floor in lowlight conditions
- Accept planned paths of patrolling in a known environment
- React to unknown obstacles
- Need to traverse multiple types of flooring common to industrial and professional settings on one elevation
- Ability to communicate the information to a controller no more than 325 ft from the platform
- The robot location must be known

1.4.4. Proposed System

The proposed system is a Sensor Suite to be added to available platforms in order to accomplish the above functions. The available platforms are a Create Robot and a RRV. It will be a developmental program that will integrate available emerging sensor technologies. The performance of the MoHSeR robotic platform will be comparable to the ability of a human sentry to identify and locate a human life form. Market research has indicated the potential to enhance performance to achieve operational capabilities outlined in this document. The proposed sensors to address the needed functions are: a pyroelectric sensor to identify a human heat signature, a microwave motion detector to determine if the heat signature is in motion and act as a confirmation to the pyroelectric when the platform is stopped, a video system to relay visual information to the operator, and ultrasonic sensors to provide obstacle avoidance. These sensors address in full the requirements as listed by the functions.

1.4.5. Sequence of Operations

The MoHSeR robotics platform will be deployed and will perform a programmed autonomous search of a planned indoor arrangement which has adequate space for the MoHSeR platform to maneuver.

- A search of the facility is to be performed to verify the presence of human heat signatures; this will be accomplished by a planned path.
- While the platform is in motion the pyroelectric sensor will scan the environment for heat signatures.
- If a heat signature between 8-15μm, the standard signature of humans, is detected, the platform will stop. The camera will record the environment, and the microwave

sensor will confirm motion of the heat signature.
- The information will be wirelessly transmitted to an operator.
- The robot will resume its routine path.

1.4.6 MoHSeR Benefits

Humans are costly and lack the ability to locate person in all lighting and environment conditions. In addition to the performance shortcomings, the use of humans put additional lives at risk in hazardous environments.

1.4.7 Capabilities Required

A table of Key Performance Parameters, or KPP, is included below in Table 1.1. These metrics serve as a foundation for operational requirements.

Table 1.1: The Key Performance Parameters for MoHSeR

System	**Requirements**	**Objective**
Sensor Suite	Identify a heat signature no less than 15 feet away from the platform. The heat signature of a human is, on average, between 8-15μm, this includes the heat signatures of other items including candles and animals. The sensor suite should differentiate between these signals. The pyroelectric, microwave motion, compass, and ultrasonic sensors must be compatible with an OOPic Microcontroller.	Identify heat source and possible intruder
Localization and Mapping	Maneuver around obstacles on the planned path of motion using ultrasonic sensors and algorithms. Discern relative location due to known environment.	To maintain systematic search patterns and high percentage performance while maneuvering obstacles. Identify location of platform accurately to relay to operator.
Transmission	All wireless communication must be designed to a range of 325 feet. No signals may interfere with each other. The camera will run on an	Relay information

	independent transmitter due to memory and frequency requirements.	
Pan	Continuously pan the forward 180°	Expand field of view
Power	It is desired that MoHSeR work in 12 hour shifts. A minimum of 8 hours is required before recharging.	Increase applicability

2. SYSTEM ARCHITECTURE

Before the individual sensors are introduced, it is important to establish MoHSeR as a system. Although more about the sensors and integration will be discussed later in the report, this will provide an opportunity to understand the overall system and how the components will interact with the Create platform.

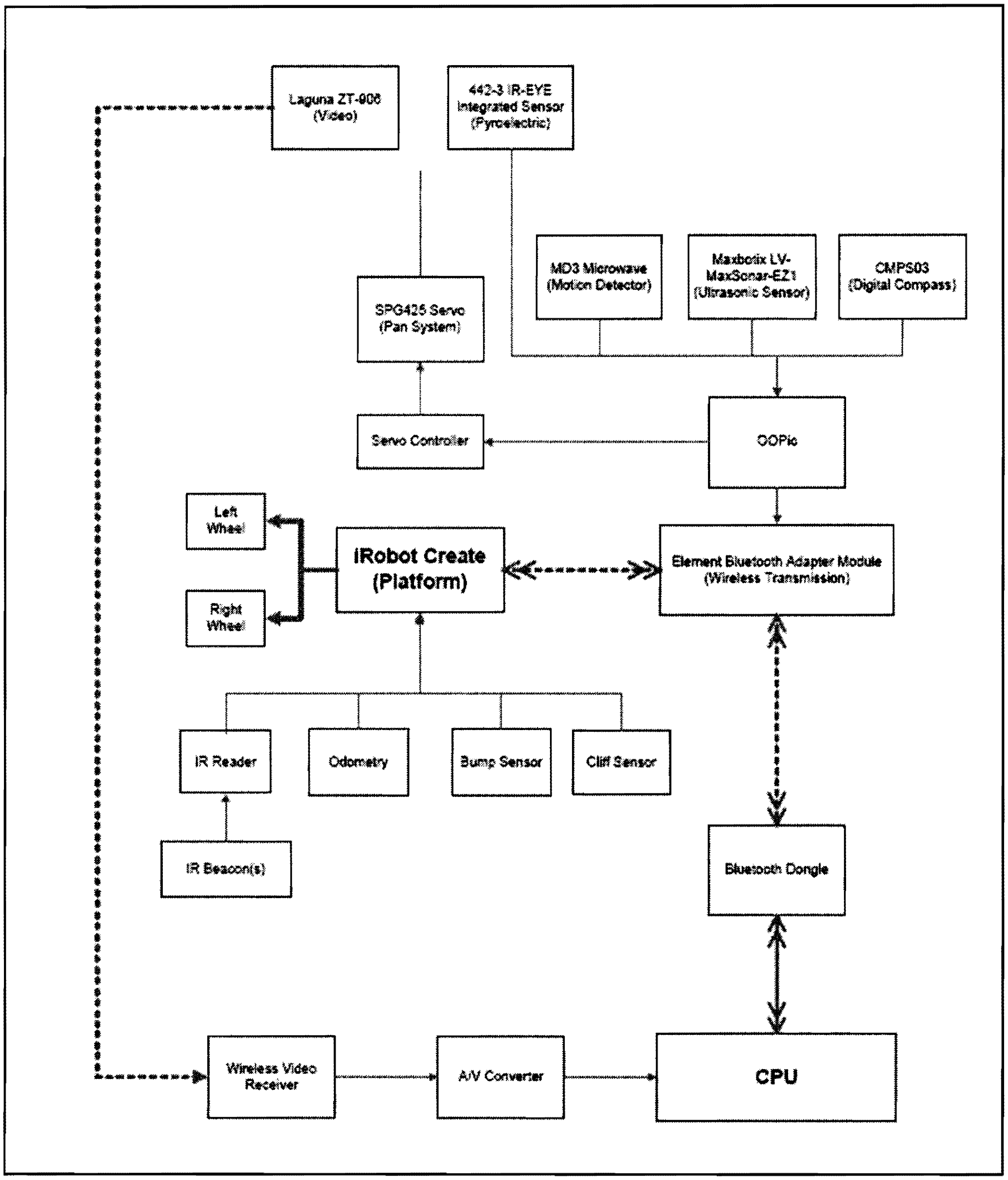


Figure 2.1: A Top-Level System Diagram for MoHSeR

Essentially, MoHSeR is designed to continuously pan two sensors, the camera and pyroelectric. The pyroelectric is panning for heat signatures, and the camera follows as a visual confirmation. The camera will directly transmit its data wirelessly to a receiver and then be converted to a digital signal to be sent to the operator. Likewise the pyroelectric will be sent to an OOPic, or microcontroller. As can be seen, the microcontroller will accept data from multiple sensors: pyroelectric, microwave, and compass. Each sensor performs a vital function. The microwave confirms or denies motion; the compass and ultrasonic sensors enable mapping and localization. Note that none of these sensors will be panned since doing so would introduce motion to these motion sensitive sensors. The OOPic is designed to perform simple onboard algorithms on data with minimal memory, so this will be designed to provide a formatted output to the BAM. The BAM will also receive localization and mapping data directly from ultrasonic sensors. The BAM directly communicates with the CPU via a dongle. From the CPU, a graphical user interface (GUI) developed with the assistance of open software, will perform several algorithms requiring more memory such as mapping.

Additionally there will be onboard sensors interpreted by the small computational power of the Create. For instance, these include the bump sensor and infrared reader.

Notice this system completely addresses the functional requirements of the project. MoHSeR will be able to relay information using Class 1 Bluetooth. This information will be collected in low light conditions, and MoHSeR will be intelligent enough for path planning. For more detail, let's now introduce each sensor individually.

3. SENSOR SUITE

3.1. PYROELECTRIC

3.1.1 Pyroelectricity

Pyroelectricity is defined as "polarization caused by change of temperature."[1] The spontaneous polarization creates a dipole moment, or charge unbalance; this dipole moment results in the creation of a current as the ions are displaced in their crystal lattice. Figure 3.1 shows a polarized and unpolarized crystal structure, and the equivalent dipole. If the pyroelectric material is held at a constant temperature for a period of time it becomes "stable", or neutrally charged. This is due to the free charge balance from the external surroundings to the surfaces of the pyroelectric and internally to the surface. A neutrally charged material ceases the current flow.

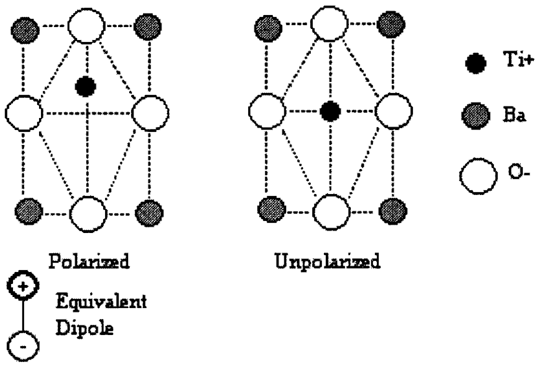


Figure 3.1: Side view of BaTiO3 crystal, polarized and unpolarized, shown with the polarized equivalent dipole.

Because there is such a large internal resistance of the material, the created current manifests itself as a voltage according to Ohm's law, $|\Delta V| = IR$, where ΔV is the change in voltage, I is the current, and R is the resistance. The change in voltage created by a pyroelectric in the circuit shown in Figure 3.2 is:

$$\Delta V = \frac{-Ap\Delta T}{C} e^{-t/RC} \quad [2] \qquad \text{Equation 1}$$

Any voltage response of our pyroelectric sensor should be predicable and easily tested. It may be possible to predict the distance a heat source is away from the robot by using this equation and mapping the ΔV response of a source. This is useful in determining location. As shown in Figure 3.3 and discussed above, the ΔV response is temporary. How long a response is seen is due to the time constant of the system, or the resistances and capacitances. One of the objectives of next semester is to test our sensor for similarities to Equation 1, and to adjust the time constant of the system for efficacy.

[2] Where A is the surface area of the pyroelectric, p is the pyroelectric constant, ΔT is the change in temperature of the material, C is the total capacitance, $C=C_L+C_x$, and R is the total resistance, $R=(1/R_L + 1/R_x)^{-1}$, where C_x is the crystal capacitance equal to ε A/a and R_x is the crystal resistance and equal to $a/\sigma \cdot A$ with σ equal to the crystal conductivity.

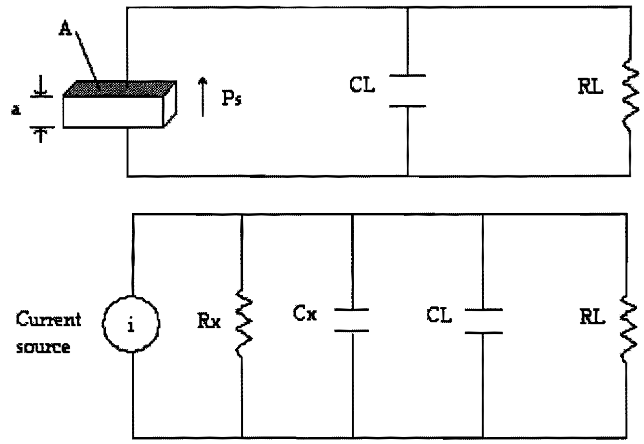


Figure 3.2: Top: Experimental setup with pyroelectric material in parallel with load resistance and capacitance. **Bottom:** The equivalent electrical circuit.

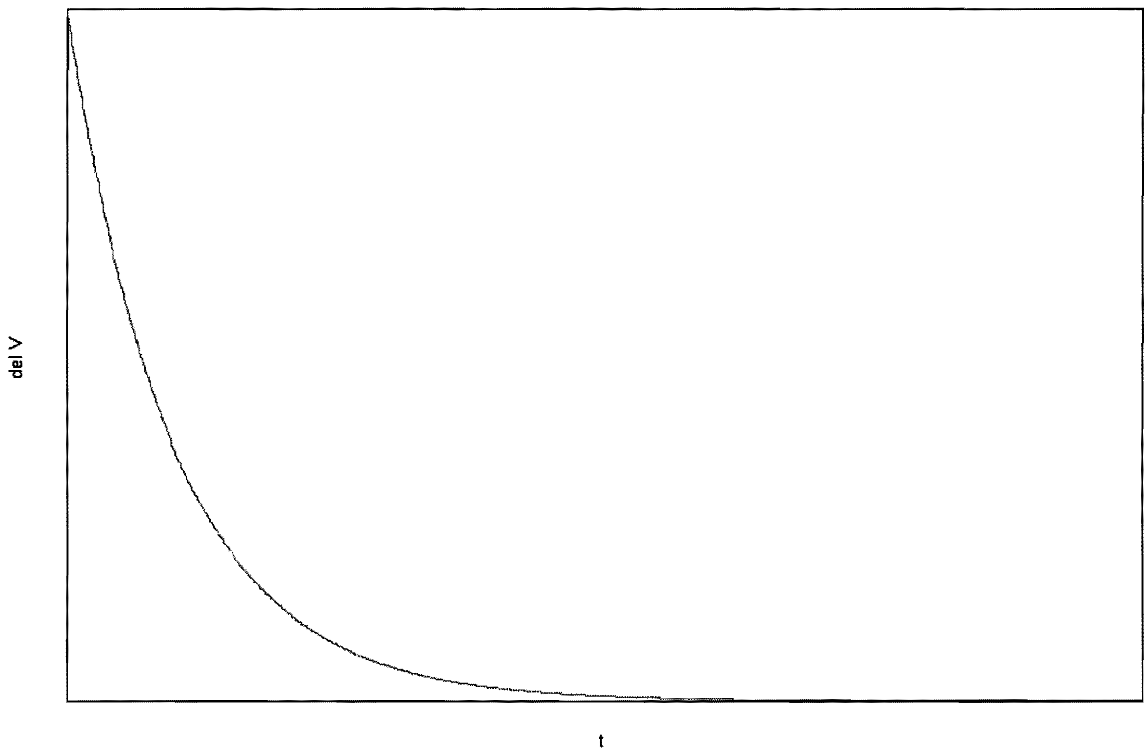


Figure 3.3: Plot of $\Delta V = e^{-t}$. This plot is meant to represent the voltage response of the PIR as a function of time, to show how the pyroelectric will respond with time.

3.1.2. Fresnel Lens

To use this type of material as a human infrared sensor effectively, a device called a Fresnel lens is added to concentrate, or "collect", the infrared waves with a much flatter, lightweight design than the thick converging lens of equal converging power. The contours of the converging lens, shown in Figure 3.4a, are approximated by "right circular cylindrical" parts and grooves, as shown in Figure 3.4b and c. To complete the setup, a cone is added to keep the pyroelectric sensor the focal length away from the lens and to reduce IR contamination. Currently, our cone is made out of heavy paper, but during testing we found that some anomalies were likely due to the Fresnel lens not being parallel with the sensor, or rather due to the cone's flawed shape. The cone will later be made from a low-IR-transmitting plastic to increase the longevity and accuracy of the cone. There are losses associated with a collecting Fresnel lens configuration. The first is loss due to reflection. This can be predicted using Fresnel's equations and a graph of reflection as a function of incident angle, Figure 3.5, is shown to illustrate that the best view is going to be less than 120 degrees based solely on this form of loss. The second loss is a result of the large width of vertical step between grooves. If the width between grooves is large, it causes light to scatter, which reduces the contrast of an "image". Or in our case, the fuzzy difference between background and source will result in a smaller ΔV. Losses are also seen from large widths of vertical step as shadows and blocking effects. This causes some rays to be blocked, which also reduces the contrast and intensity of the "image". [2] These losses are the limiting factors associated with the use of a Fresnel lens. The view seen from a normal converging lens is close to 180 degrees, the view seen from our Fresnel lens was tested to be about 15 degrees. For our application, this is not a restrictive problem and can be resolved during integration simply by panning the pyroelectric and Fresnel lens setup to see the forward 180 degrees around the robot.

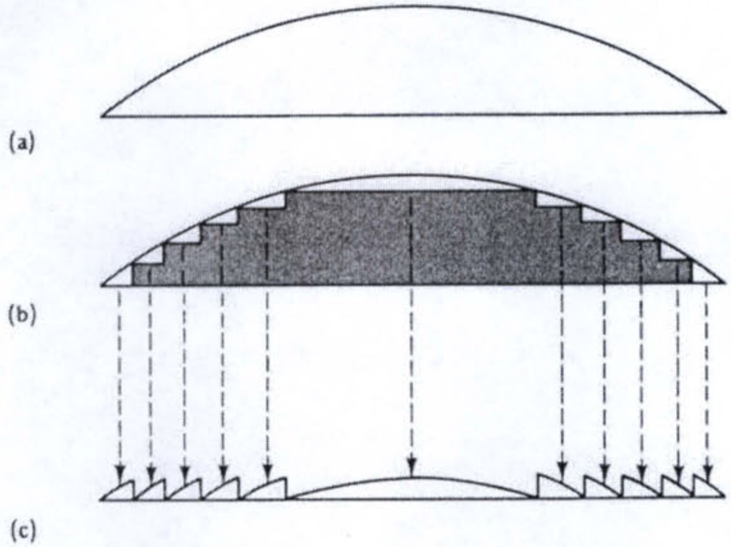


Figure 3.4: a) Thick converging lens. b) Parts of the lens removed and

c) the resulting Fresnel lens. [3]

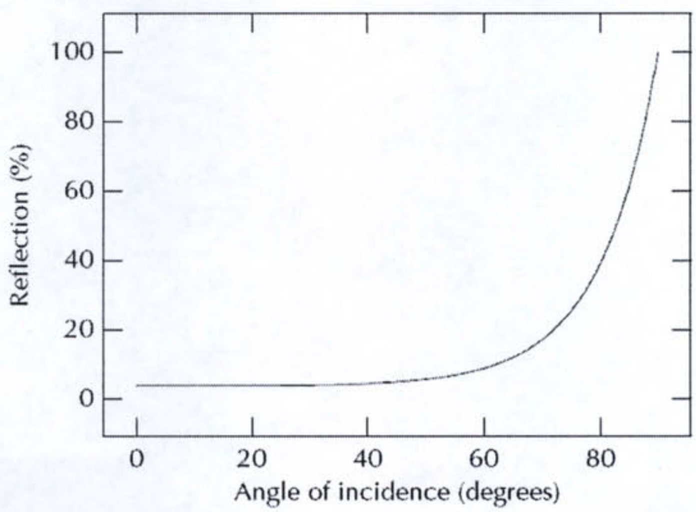


Figure 3.5: %Reflection vs. incident angle for a collecting Fresnel lens.[2] As incident angle increases, the % reflection increases. For this configuration, collecting, angles above 60 degrees will result in significant losses.

3.1.3 Sensor Solution

The human body has an infrared heat signature between 8 and 15 μm. Pyroelectrics offer a wide range of infrared wavelength detection and one whose spontaneous polarization reacts best to infrared wavelengths from 8 to 15 μm can be chosen. A Pyroelectric Infrared Sensor (PIR) was chosen as our infrared detector over infrared cameras and photoconductors because it is able to inexpensively, efficiently, and straightforwardly detect a human infrared source in low to no light conditions. Pyroelectric sensors are relatively cheap, the only real costs being material and polarizing process. Because the sensors are small, material costs are minimized and their light weight will not contribute significantly to the robot's payload.

Specifically, the Eltec 442-3, a Lithium Tantalate PIR, was chosen for human infrared detection. The sensor and Fresnel lens were purchased together for less than $75 from Acroname Robotics. A single 9-volt battery is needed to run the sensor, and it was tested to have at least a 13 degrees by 20 feet range of detection, with a voltage response near 5 volts in a dark room. Eltec does not declare the sensor's infrared detection band, but it is stated that candles and pets can be detected in addition to humans. Detecting non-human sources cannot be avoided with this sensor, but the certainty of human detection can be increased with the addition of other sensors such as a motion detector. Room temperature is included in its temperature range of –40 to 70 degrees C. A full list of features and operating characteristics can be seen at the Acroname website, http://acroname.com/robotics/parts/R1-442-3.html.

Figure 3.6: The Pyroelectric Sensor

3.1.4. Sensor/Robot Interface

To interface with MoHSeR's wireless transmitter and programming, a microcontroller from OOPic will be used. To leave the analog I/O ports on the wireless transmitter free for other sensors the OOPic will be used as a yes/no filter, and thus take a digital I/O port instead. The OOPic microcontroller will take the voltage produced by the pyroelectric in an analog input, decide if the voltage is higher than the tolerance set for a no-source situation, and send a "yes" or "no" via a digital output to the wireless transmitter, and thus the computer's programming, where a "yes" will indicate an infrared source has been detected. Figure 3.7 shows how Acroname suggests the PIR should be interfaced with the OOPic. A voltage modulator is added to modulate the baseline voltage.

As discussed in section 3.1.2, panning the PIR to see a more complete view around the robot will be implemented. The PIR will be mounted to a servo atop the MoHSeR platform. It was discussed whether or not a tilting motion should be added in addition to panning to better view intruders. The FOR's state the robot should be able to detect a human standing on the ground. At 20 feet, the maximum distance tested to date, the robot will see about 3 feet above the ground. Though the detection of a human is most effective when the pyroelectric passes along a human's chest, the most radiative part, tilting was deemed unnecessary through testing. While testing, a large enough voltage response was seen at 20 feet, meaning human legs wearing jeans are radiative enough to detect.

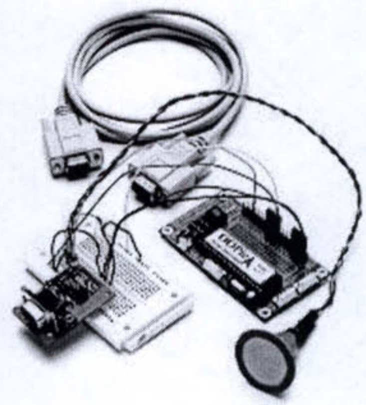


Figure 3.7: The Component Setup for the Pyroelectric.

3.1.5. References

SEE APPENDIX

3.2 MICROWAVE MOTION DETECTOR

3.2.1 Overview/Method of Operation/Technical Status

The MoHSeR robot can detect infrared heat energy in the expected range for a human being. However, it can also detect non-human heat sources, such as a flame. A microwave motion detector will be used to verify the findings of the pyroelectric sensor.

The microwave motion detector incorporates a lot of interesting principles, despite the fact that its output is effectively a simple on-off switch. When it is turned on, the device oscillates electrons back and forth along its antenna a billion times a second. When these electrons move back and forth, they are not isolated from their environment. The electrons will pull other charged particles with them as they move back and forth. In this way, information is transferred through the air. These electromagnetic waves travel in all directions, as a ripple does in a pool of water. When the waves encounter an object, they will reflect off the surface of the object, scattering in all directions. Some of these waves will return to the original point of transmission. These fragments are sensed by the antenna and "analyzed" within the circuit.

As was mentioned earlier, the broadcast frequency (or the frequency at which the electrons oscillate along the antenna) is 1 GHz. Assuming the motion detector and its surroundings are stationary, the reception frequency (or the frequency at which the electromagnetic pulses return to the antenna) should be the same frequency. However, if an object is moving, its frequency WILL change. Suppose two pulses are sent out towards an object moving away from the transmitter. The first pulse will hit the object at one distance and reflect backwards. However, the second pulse will reach the object at a greater distance than the first pulse did. The object moved farther away between pulses. Thus, the time between the pulses will be greater. By the same token, the frequency will be lower than the broadcasting frequency, since the pulses become farther apart (and less frequent). The reverse is true if an object is moving towards the transmitter – the pulses are closer together, and thus the reception frequency will be higher.

So, the motion detector notices motion by noticing a change in frequency. If this change in frequency is above a certain threshold (which can be adjusted on the motion detector itself), the motion detector will send out a signal, which activates an electronic switch and tells the microcontroller that a change is detected.

3.2.2 Hardware Requirements

The specific motion detector model we will be using is the Ramsey MD3 Microwave Motion Detector and can be seen in Figure 3.8 (available at http://www.hobbytron.com/R-MD-3.html). While the unassembled version of this motion sensor can be obtained for $34.95, in the interest of time and simplicity we will be sending away for the preassembled version, costing $49.95. This is still reasonable given the motion detector's capabilities – it has a range of ten to twelve feet, adjustable sensitivity, portability, a relatively small footprint, and appears to suit our needs. It can even be made directional by putting a small metal plate behind the motion detector, which should focus most of its attention forward (and thus hopefully help focus the detector on what the pyroelectric sensor sees).

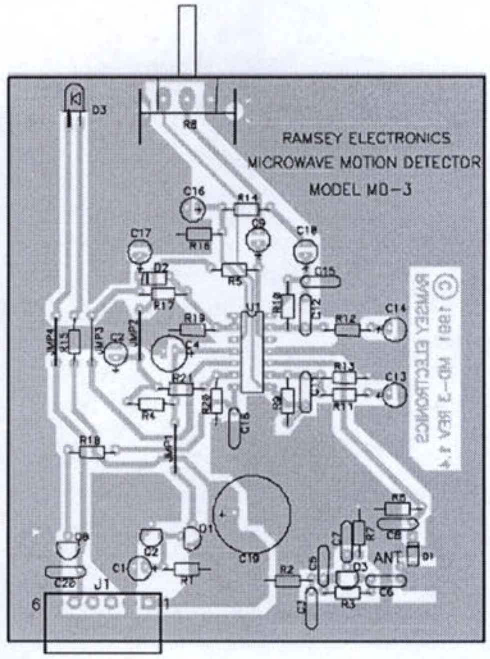


Figure 3.8: A Diagram of the Microwave Motion Detector

3.2.3 Input/ Output

This motion detector is fairly self-contained – it can run off of a 9V battery. It also has a very simple output. As mentioned above, the motion detector simply supplies a "high" or "low" voltage on its output pin, depending on whether it detects motion or not. A "high" voltage (approximately +5V) can be used in many situations, including switching a transistor on, which could operate a much higher power circuit. However, not much power is necessary. A transistor can be used to send a signal to a microcontroller to activate the proper programming routine, and might even be able to send the raw signal straight to the microcontroller, depending on how powerful the output actually is. This can be determined from experimentation.

3.2.4 Software Requirements

Using a microcontroller to read the detector's output has a number of advantages. It should simplify programming somewhat, but it also gives MoHSeR the option to ignore the motion detector entirely. This is important, because MoHSeR is a mobile robot, and the motion detector will be onboard. If MoHSeR moves, the distance between the sensor and its surroundings (a wall, for instance) will change with time. This may set off the motion sensor prematurely, leading to annoyed controllers and a potentially useless robot. Instead, MoHSeR will only listen to the motion detector at the appropriate time – when MoHSeR has come to a stop (whether planned or due to the pyroelectric sensor detecting something). This will minimize the background noise of the robot's motion, increasing the accuracy of detection.

3.3 VIDEO

3.3.1 Overview

As outlined in the functional requirements of this project, the robot must be able to communicate to a controller. Likewise, the solution must be able to operate in low-light conditions and assist in identifying the heat signature of interest. Many technical solutions are viable to accomplish these tasks; for instance, the application of an olfactory sensor could identify the presence of a person, yet this type of sensor would not greatly assist in allowing the operator to identify a person of interest. As determined by research, the most applicable solution is the implementation of a camera. This solution provides many advantages: First, the controller will be able to monitor real-time video of the area under surveillance during normal operation; this provides added functionality to the project. Second, cameras provide recordable documentation for identification and prosecution of any presumed intruder. In all, a camera provides a level of reliability for the controller. Certain cameras are able to operate wirelessly and efficiently in low-light conditions. The remainder of this documentation will consider the visual sensor system as a solution to the MoHSeR project.

3.3.2 Technical Status and Considerations

Inherent in applying a camera as a solution to this project is to thoroughly understand the sensor and the system. There are several open-source video solutions applicable to robotics, but the MoHSeR project is unique. As in any design there is a balance of what is needed, desired, and required. The final design choice for this sensor was analyzed on multiple criteria. First and foremost, the sensor solution must fulfill the functional requirements. Second, the sensor must be able to integrate with the sensor suite as seamlessly as possible. Third, the solution must be within the constraints of the time and financial resources.

In order to ensure the best solution for MoHSeR, research on several solutions was conducted. For instance, many of the open-source solutions applied an embedded vision processor named CMUcam3. Upon first examination this solution was appealing. The CMUcam3 is a fully programmable embedded computer vision sensor, meaning with the correct setup and software this camera could perform however desired. The resolution was sufficient at 352x288 pixels, and it recorded 26 frames per second (FPS). Additionally, since the camera was highly compatible with C code, the ability to grab frames and perform data collection was impressive. However, there were several drawbacks to this solution and similar video-sensor solutions. First, the CMUcam3 was a hardwired solution. In order to fulfill the wireless nature of the platform motion, the CMUcam3 would require the development of a secured network on which to operate. Although setting up a network has been done, the monetary constraints of the project did not behoove this solution to be accomplished adequately. Second, the CMUcam3 was designed to operate in daylight conditions. The functional requirements directly state that MoHSeR must operate in low-light conditions. In order to make the CMUcam3 operable in these low-light environments, either an infrared sensitive lens would need to be added or a light would need to be added to the sensor suite. Both of these solutions were impractical since MoHSeR needs to be the least intrusive as possible to its environment and the customization of the expensive CMUcam3 was neither confirmed nor advised by experts. Thus the CMUcam3 was ill-advised and did not perform all of the functions necessary for the project.

Further research led to a more traditional surveillance solution. From anecdotal evidence, it seems the majority of roboticists that have attempted creating a wireless night vision system encourage the use of traditional surveillance cameras. Thus, the primary focus of this solution built upon this founded knowledge to find appropriate hardware.

Additional Design Considerations

Special considerations were given to finding the characteristics: minimize illumination, maximize resolution (with a minimal of 320-video line – this is the least resolution for recognizing forms in motion), high compression, 2.4GHz or 300 ft transmissible range, and maximized night vision range. In addition, the camera must be lightweight since the Create can only maintain a 5 lb payload.

3.3.3 Hardware Requirements

3.3.3.1 Camera

Figure 3.9 displays the sensor that will be applied as the solution. This camera is an advanced day and night wireless video camera; it is manufactured by Laguna[3]. This device, henceforth referred to as Laguna, is applied in remote monitoring situations and transmits real-time data to a monitor. The camera is capable of transmitting data over a 2.4GHz frequency at a range of 325 feet. Inherent in this solution are several advantages and disadvantages. Primarily, this solution provides the established technical hardware developed in standard surveillance applications. For instance, the camera has the capacity to view in night vision up to a range of 49 feet. The minimum illumination is 1.0 lux. This is significant since illuminance is the total luminous flux incident on a surface per unit area. In common terms, illumination is a correlation between the intensity of light and human perception of brightness. Thus the lower illumination a camera can detect the better a low-light camera it is. The 1.0 lux specification is an acceptable amount for MoHSeR since this is equivalent to near darkness.

[3] Part Number: ZT-906T

Figure 3.9: A Low-Light Wireless LED Camera

Addressing the other design considerations, Laguna is only 0.63 lbs and it has a resolution of 420-video lines. In terms of pixels this is equivalent to 512x582 which is sufficient to relay clear images for identification. The disadvantage to this solution is the loss of the ability to customize; when looking at MoHSeR's requirements, this is not a difficult compromise considering other solutions did not fulfill the fundamental requirements within the project limitations.

It is important to note, the camera requires additional components to make the system completely operational. These are discussed below.

3.3.3.2 Wireless Receiver

The recommended receiver for the above camera is the Laguna ZT-707. This is a four channel receiver that outputs analog audio and video components. A figure of this hardware is shown in Figure 3.10. This solution is appropriate for MoHSeR since this 2.4GHz signal will not conflict with Bluetooth; this is because Bluetooth cycles through several frequencies per second. The interference when they are coincidentally on the same frequency will be negligible. This will be confirmed through experimentation.

Figure 3.10: The wireless receiver broadcasts at 2.4GHz but does not interfere with Bluetooth.

3.3.3.3 Analog to Digital Converter

Since the output from the receiver is in component A/V format, the system needs the application of a peripheral component interconnect, or PCI. The PCI is hardware that works as a digitizer and multiplexer. Thus it is imperative that this output be acceptable with the overall objective and system for MoHSeR. The solution is the DIGIVUE EDV-XV425 PCI. This hardware is displayed in Figure 3.11. Notice that the important information from this hardware is the fact that it digitizes at a rate of 60 frames per second (FPS). More importantly, the hardware automatically compresses its MPEG4 output claiming that for every 1 week of recordings only 7GB of memory are required. This is more than adequate for MoHSeR. Additionally, this hardware comes with a prepackaged software interface that eases the application of this product and provides event recording modes.

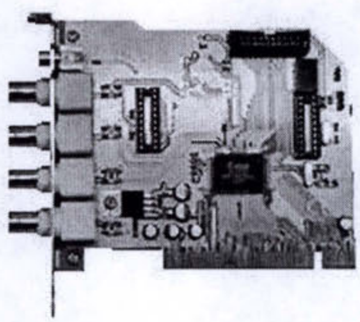


Figure 3.11: The PCI required to convert the analog signal into a digital output

3.3.3.4 Operational Modes/Software Requirements

The camera will relay real-time data at all times to the controller. It is desired, but not necessary, that the data be recorded when triggered by the motion detector and/or pyroelectric. This determination will be made experimentally. Software is included with the PCI to provide remote viewing and event recording. These are both ideal for MoHSeR.

3.3.4 Specifications/Documentation

The manufacturer's specifications for the camera are attached in the appendix. It is notable that an added benefit to this solution is a viewing angle of 39°.

3.4 OBSTACLE DETECTION

3.4.1 Scope

The operating requirements for locomotion and navigation are that the robot be able to have an understanding of its location. Likewise, it must possess the ability to maneuver unexpected path obstructions.

3.4.2 Technical Status and Considerations

Inherent in using odometry for navigation, it is necessary to minimize the errors which can be introduced by jolts or bumps. The current obstacle detection on the Create is a bump sensor which by its very operation produces these accuracy reducing bumps. It was concluded that this bump sensor is inadequate for detecting path obstructions without affecting the integrity of the robots localization data. It was then decided that some sort of proximity sensor would need to be applied to be able to acknowledge obstructions before impact.

The present electronics already incorporate the bump sensor. The issue is that the bump sensor is really just a switch that closes a circuit when it bumps into something. In order to hook the proximity sensor in parallel or to replace the bump sensor it is necessary for it to perform the same switch type output. While this proximity switch has more complicated hardware, thus more expensive, it would potentially save us the more complicated task of programming. In addition, it will provide an increased assurance in the software system that exists for the bump sensor.

An additional possibility is to have a completely separate system for the proximity sensor was evaluated but it would require the use of the OOPic micro controller which is processed remotely on a CPU. While this option is more analogous to our project goal of an all encompassing sensor suite, this lag with the remote processing of the OOPic attachments could potentially cause a pause that would allow the robot to contact the sensed obstruction. The final consideration is that of the specific type of proximity sensor to employ on the robot. It is necessary for the sensor to not interfere with any of the other sensors on the robot. It must also be capable of detecting all possible obstacles.

3.4.3 Obstacle Detection Solution

The first proposed solution system is a combination of the sensor switch and a simple sensor. Because of the unknown performance speed of the signal turnaround for the information sent through the OOPic, and due to the fact that failure in this performance speed to relay the signal would result in the failure of purpose for this sensor system. It was determined that a parallel system would need to be wired directly into the bump sensor circuit. The same sensor will be used but the signal will be split going to both the OOPic and to a small voltage switch which will be put in parallel with the bump sensor. It has since been determined that the mechanism of the bump sensor would also be transmitted through the Bluetooth connection and therefore holds no processing speed advantage to the OOPic sensor data transmission. As such the LV-MaxSonar-EZ1 sonar range finder will be mounted on the top of the bump sensor and will be wired to the OOPic for data acquisition.

3.4.3.1 LV-MaxSonar-EZ1Sensor

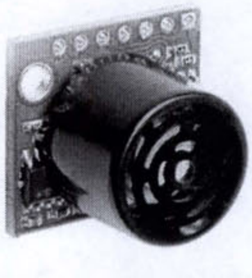


Figure 3.12: LV-MaxSonar-EZ1 high performance sonar range finder

The chosen sensor is the LV-MaxSonar-EZ1, shown above in Figure 3.12. The audible sound this sensor uses to determine distance will not affect any of the other sensors being used on the robot, in addition to being able to sense any possible obstruction the robot may encounter. Other benefits include the range of 0 to 254 inches, as well as a variable power input of 2.5 to 5.5 Volts. This wide range of detection will allow the robot to be programmed to any level preventative avoidance do to the potential for very early detection. The variable voltage input will enable the EV1 to be power from any number of the preexisting power supplies already on board the Create.

The beam capability of the sensor is very well suited to our application, it is narrow and far reaching which allow for increased range without the interference of obstacle detection that may be outside of the direct path of the Create. Figure 3.13 bellow represents the claimed beam characteristic for the EZ1 on a 12-inch grid.

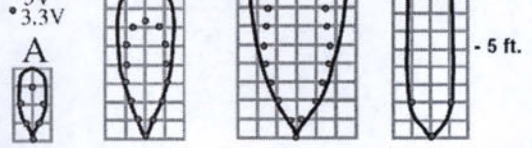


Figure 3.13: MaxBotix provided beam characteristics. A is the detection of .25-inch diameter dowel. B is the detection of a 1-inch diameter dowel. C is the detection of a 3.25-inch diameter rod. D is the detection of an 11-inch board parallel to the sensors front.

While the EZ1 sensor has three different forms of output, TTL serial, PWM, and analog, the pulse width modulation output will be the utilized mode for data acquisition. Therefore the actual connections to the sensor will be to the power, ground and PW ports, as shown in Figure 3.14 below.

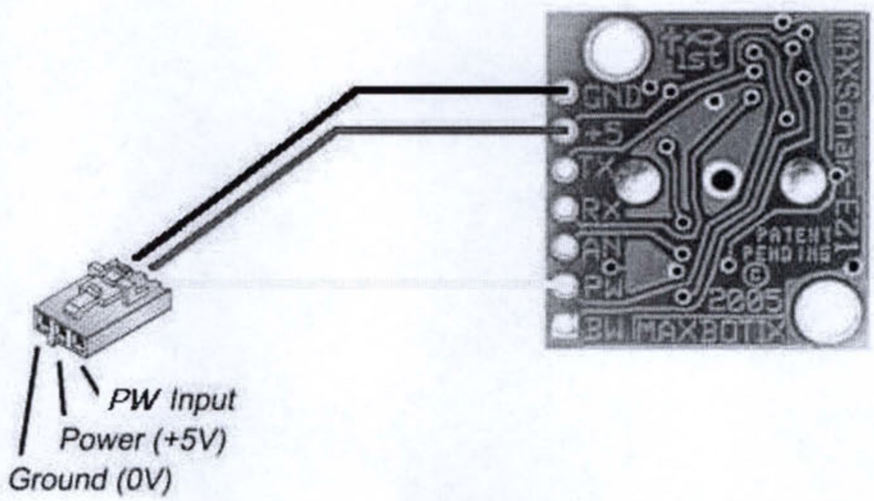


Figure 3.14: PW connection requirements to the LV-MaxSonic-EZ1 sensor

In pulse width modulation mode the sensor will continuously be in run mode. It will sense the range every 49 milliseconds and output a pulse width representation of range. The distance can then be calculated using the scale factor of 147uS per inch. When the 49mS period starts the EZ1 sends thirteen 42 KHz waves, after which the PW pulls high. Upon detection the PW then is pulled low. If there is no object detected then the PW pin will be pulling high for the entire 37.5mS duration. The rest of the 49mS is used to adjust the analog output which will not be used. This direct range reading output will allow for quick data processing. One additional consideration is that due to the acoustic nature of the sensor it would be suggested to restrict the number of sonic sensor to one in order to prevent false readings or noise interference.

3.4.3.2 OOPic

See 5.2.1 OOPic

3.5 TRANSMITTER

3.5.1 SCOPE

Part of the requirements of the robot is to be able to communicate wirelessly with a base computer to give the robot's status and alert the user of potential intruders. The transmitter needs to be able to transmit over a reasonable range and have the capacity to send the data formats of the sensors. There are a number of ways to achieve this, such as using Wi-Fi.

3.5.2 BLUETOOTH

Bluetooth is a communications system that is used to communicate wirelessly between two devices. The additional benefit is that it is extremely low-powered compared to Wireless LAN. Like Wireless LAN, it operates in the industrial, scientific and medical (ISM) unlicensed radio band of 2.4 GHz. However, it uses a technique called spread-spectrum frequency hopping that prevents interference with other wireless devices. The Bluetooth device will use randomly chosen frequencies within a defined range, changing on a regular basis at 1,600 times per second. Currently, a Class 1 Bluetooth device has the highest range, capable of communicating to approximately 300 ft. However, Bluetooth is not the perfect device as it is prone to packet errors and random disconnections. While reliability is questionable, it is convenient since any computer with Bluetooth can wirelessly connect to it.

Element Direct Bluetooth Adapter Module (BAM) will enable wireless communication between the Create and a personal computer. The BAM was designed to work with Create; therefore, it easily attaches to the expansion port in the cargo bay using a DB-25 male connector bypassing any need for extra wires or cables. With Serial Port Profile (SPP) support, the BAM appears to the host computer as if it was connected directly to its serial port. The BAM's I/O Connectors exposes the power and 20 user-programmable I/O pins which allows for convenient access to integrate sensors and actuators.

Figure 3.15: BAM

BAM's connector pin.

CB	Function	BAM
23	Low Side Driver 1	LD1
22	Low Side Driver 0	LD0
20	Digital Output 2	DO2
7	Digital Output 1	DO1
19	Digital Output 0	DO0
6	Digital input 3	DI3
18	Digital input 2	DI2
5	Digital input 1	DI1
17	Digital input 0	DI0
4	Analog Input	AIN

BAM	Function	CB
LD2	Low Side Driver 2	24
Vsw	Switched Battery Voltage	10, 11 & 12
Vpw	Battery Voltage	9
+5V	Switched 5V	8
Gnd	Ground	14, 16, 21 & 25
PT	Power Toggle	3

Figure 3.16: The I/O of BAM

3.5.3 TECHNICAL STATUS

The BAM and the Bluetooth dongle have been purchased and is awaiting setup and testing. XBee will only be implemented if we cannot find a reliable solution using BAM.

4. PLATFORMS

4.1 CREATE

4.1.1 Introduction

The Create robotic platform is the primary platform for this project. It has independently controlled wheels, wheel encoders, an IR detector, bumps sensors, battery temperature and charge sensors, to name a few features. Several demonstrationss are stored on the Create's internal PIC. More advanced control requires connecting a controller to the Create's open interface through its TTL serial connection. This can be accomplished by connecting a controller directly to the TTL Tx and Rx or by using a transmitter to create a virtual serial connection with a PC. The advantages and disadvantages of each setup are discussed below in the Interface section. The Create's suggested load capacity is 5 pounds. The Create comes with the basic hardware components of a robot. In order to increase the capabilities of the Create to meet the requirements of this project, additional sensors are required, particularly for improving navigation capabilities, which are discussed next. The iRobot Create is seen in Figure 4.1.

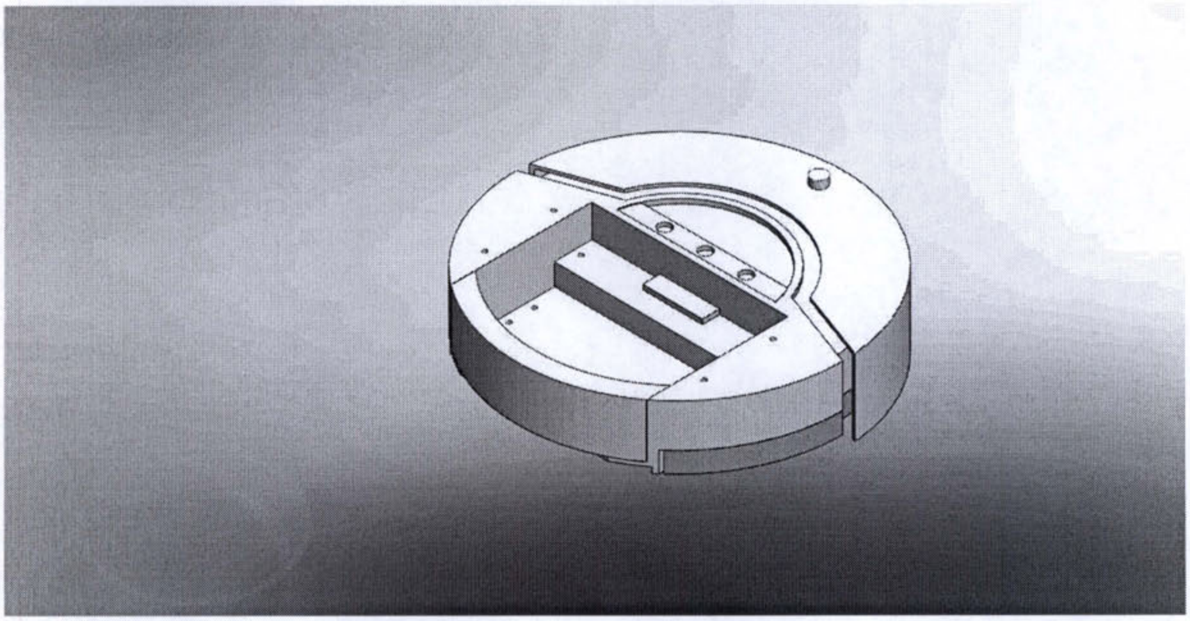


Figure 4.1: The Create Platform

4.1.2 Navigation

4.1.2.1 Scope

Integral to robot navigation are mapping, localization and path planning. These problems present the following questions: What does the environment look like? What is the location of the robot relative to this environment? How does the robot get from point A to point B in this environment? This section discusses how this project addresses these fundamental questions using the iRobot Create platform. The feasibility of Simultaneous localization and mapping (SLAM) is discussed next.

Simultaneous localization and mapping (SLAM) builds a map of the environment while locating the robot within this map. It is a robust technique for creating foundation of an autonomous robot and has been developing steadily in recent years. SLAM requires memory for storing the map, processing power for using SLAM algorithms, software for integrating the sensor information, and sensors. The amount of memory required depends on the size of the environment and the resolution of the map. The requirements for processing power and software depend on the type or types of sensors being used. Accurate and robust sensors enable more efficient computation. Many applications of SLAM use the SICK laser. It has excellent range and accuracy. The SICK laser is incompatible with the Create because it is too heavy and because of budget constraints. SLAM sometimes uses just sonar sensors. This solution would require at least six sonar range sensors, additional hardware (I/Os) to transmit the senor data to the PC, software for using

SLAM with sonar, and a considerable effort for implementation. A disadvantage of Sonar range finders is that they have limited range and precision. This results in inefficient computation to correct for uncertainties. Sonar has problems with false data association and divergence.

Some applications of SLAM use a camera and sonar sensors. The use of a camera is called visual-SLAM or vSLAM. One disadvantage of using is camera is that is has a low frequency of update compared to the other common options for SLAM. The camera and sonar complement each other. The camera corrects the data association problem of sonar sensors, and sonar corrects the cameras low update frequency. In addition to the requirements for sonar, vSLAM would require software to integrate the sensor information together and to fuse the sensor information using SLAM. This software could either be built from scratch or the software might be purchased or found online. If the software were found, it would still require drivers to interface the sensor information. The drivers could written from scratch, or ideally, found on the internet. Software exists that can use even a low quality web camera to implement vSLAM. Unfortunately such software could probably not be purchased due to budget constraints. Also the camera requires good lighting conditions. This is incompatible with our FORs which says that the robot must be able to operate in low light conditions. Note that there is a chance that SLAM would not work using a remote PC as the controller due to the time lag. If that was the case, it would require an onboard microcomputer with moderate memory and processing power. This is unlikely for sonar because of the small amount of data being sent, but there is a slight chance it might cause problems with vSLAM.

The more effective implementations of SLAM are incompatible with the hardware constraints. Implementing SLAM using a sonar ring was discussed with Dr. Lynne Parker. She said that even if the appropriate software was obtained, it would probably take at least a semester to implement, and that this implementation would be better suited to a computer science major. The biggest barrier to using SLAM is not getting the information to the computer but the software required to integrate and fuse this information. It is worth noting that internet searching uncovered several attempts to implement SLAM with the Create, but none succeeded. In short, SLAM makes a robot more intelligent and autonomous, but SLAM might not be compatible with the given time constraints. The approach is to first develop a solution for navigation that will work, and then, develop SLAM as a more robust option. An alternate solution, starting with mapping is discussed next.

Mapping is the problem of integrating sensory information to map the environment. For this project it is reasonable to assume that the map is known a priori. Thus, the majority of the time the robot will be following a predefined path. However, it is possible that an obstacle might accidentally be left in the path of the robot. Thus it is required that the robot can navigate around the obstacle. This is discussed further under the obstacle avoidance. The problem of localization is addressed next.

Localization is the problem of estimating the pose of the robot relative to the map. If the initial location is known, then it is called pose tracking. In this project, it is reasonable to consider the map to be 2D. Thus, pose is uniquely identified by three components: an x-coordinate, y-coordinate and orientation. The Create uses high resolution wheel encoders to track its pose; however, slight errors will quickly compound until they are unacceptable. Thus it is essential to use landmarks to minimize errors in pose. One method is to use IR beacons. When the robot detects an IR beam, it outputs a byte number to its IR signal. If the IR beacon is placed perpendicular to the y coordinate, then the robot will intersect the beam at some known y. The value of the IR signal can be used to correct the robot's y-coordinate. The same method can be used to correct the robot's x-coordinate. A diagram is seen in Figure 4.2. Now the technical details of how an IR beacons will be described.

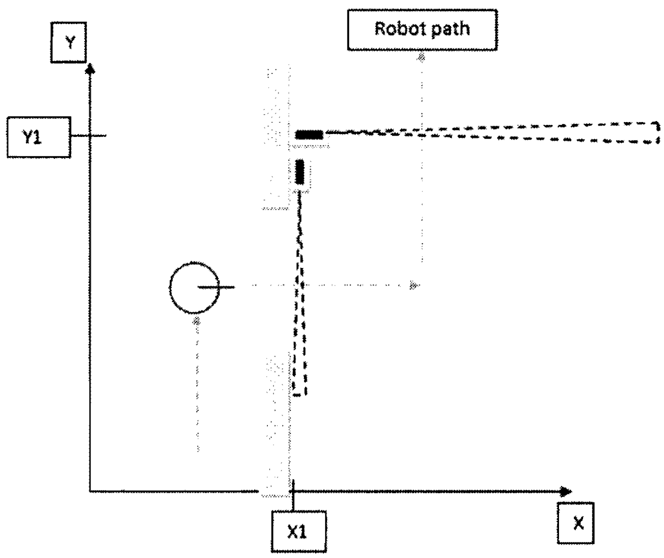


Figure 4.2 Mapping with IR Beacons

The IR beacon sends a serial command to the Create via the Create's IR scanner. The command tells the Create to output the specified byte number for its IR signal. The command is serial and consists of an "opcode" and a byte number. The "opcode" tells the Create what command is being sent and the byte number specifies which option for that command is to be executed. Since the Create uses an internal 8bit controller, the maximum number of bytes and thus options for a command is 256. For example, if the IR beacon was programmed to have the Create output an IR byte number of 243, then the command sent by the IR transmitter would be [141][243]. The 141 is the "opcode" corresponding to outputting an IR signal and the 243 specifies the byte to output. Specific coordinates would be tied to each byte number. This will set the robot's position to the coordinates corresponding to the waypoint for that IR beacon byte number. Finally the components the IR beacon are laid out.

The IR beacon consists of a PIC, an IR LED, and a 3V power supply. Each IR beacon would specify a unique byte number. The IR beacon can be seen in Figure 4.3. The beacon's circuit schematic can be seen in Figure 4.4. Note that the IR beacon repeats at the same frequency, 200ms, as the Create's docking station, virtual walls, and remote. The primary advantage of the IR beacons is the simplicity of the solution. It mainly uses existing features of the Create. Also it requires very little programming for implementation. There are some limitations of using IR beacons. IR beacons are artificial landmarks and thus require some modification to the environment. The IR beacons do not increase the intelligence of the robot. They just provide information about the environment. A robot is not really autonomous if it requires a controlled environment. Also the update frequency is low so the Create has to navigate reasonable distances between beacons. A requirement for this is discussed next.

Figure 4.3: Infrared Beacon

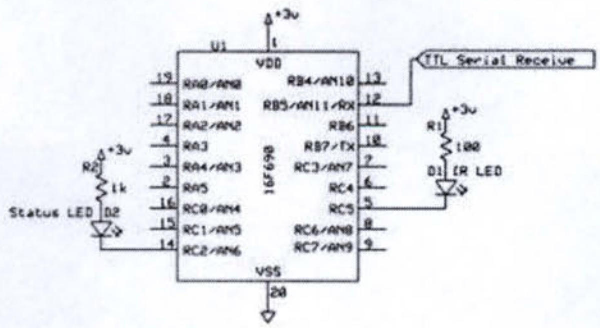


Figure 4.4: Infrared Beacon Schematic

The Create requires an accurate heading. This requires that errors in orientation be corrected. One solution is to use a wall follower. This requires several range sensors. The disadvantage is that robot must always be close to a wall, and there is a chance it might have problems crossing open doorways. Another solution is using a digital magnetic compass. The compass would constantly update the orientation of the robot. It uses the earth's magnetic field to detect direction. It communicates with an OOPic via I2C connection which has an output range of 0 to 3599. The OOPic would then convert the signal to an analogue output so that it could be sent to the PC via the transmitter. One disadvantage of the compass is that it is susceptible to metal, such as motors, rebar, etc. Thus there is a chance it might not work if it is placed near the Create's motors or the structure contains a lot of rebar. Also the accuracy of the compass might be degraded when it is converted and sent as an analogue signal. The Compass can be seen in Figure 4.5.

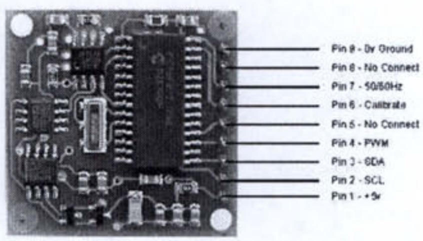


Figure 4.5: The Electronic Compass

In summary, SLAM would make the robot more autonomous. The approach is to develop a working solution to navigation given the time and budget constraints, and then focus on developing a more robust solution given the time and hardware constraints.

4.1.2.2 Technical Status

A vendor for the Devantech compass has been identified and is awaiting an order. Price is $60. A vendor which sells the beacons with preprogrammed byte numbers has been identified and is awaiting an order. Price is $5/ea. For more information, including the methodology for connecting the compass module to the microcontroller please see the appendix.

4.1.3 Interface

The Create has an open interface accessed by serial communication. The serial communication format is TTL which uses a range of 0 – 5V. The Create has its own "machine language". Commands are identified by their "opcode" which is the first part of the command and is followed by appropriate number of bytes which specify the parameters for the command. All the codes are specified in the Open Interface Manual. The Create has 32 internal sensors, e.g. actuators, bump sensor, cliff sensor that are accessed using these commands. A higher language such as C or C

++ is used for programming, and then this code is compiled into machine code and sent to the Create. It is possible connect a controller directly to the interface via the TTL Tx and Rx or use a transmitter to create a virtual serial connection with a remote PC. The advantage of a direct connection is that it ensures real-time information. If a transmitter and PC are used, a time lag is introduced which might cause problems.

4.1.4 Adaptive Mapping and Path Planning

The effectiveness of the robot to perform the specified task of complete area observation coupled with efficient maneuvering of the environment lies heavily in specifics of the robot's route and path choice. For this reason an adaptive mapping scheme will be applied both before and during the operation of the robot.

In order to keep the simplicity of operation and programming the map will be an array of squares that are of the Create's outer dimension in width. The ability of the Create to perform 90º turns without forward advancement will be used to move directly to the squares on the left or right of the robot when necessary without excessive complexity.

Once the virtual grid has been determined including all walls and permanent structures a path will be manually chosen in order to reach a user specified waypoint where the Create will stop and scan. The Adaptability of the map is applied when an obstruction in sensed by the forward sonic sensor. Upon obstruction observation the Create will advance to the forward most grid square that it can fully occupy without impacting the obstruction. The next forward square will be marked as impassable on the map, at which point a short obstacle maneuvering program will be applied. The program will evaluate both left and right hand turns. The priority will be giving to the side which the map shows as having the most grid squares in that specific direction row. The robot will rotate to this direction and proceed forward. Note that the forward sonic sensor is always on run and will not allow the robot to advance if it cannot fully occupy the next grid square. If the robot turns and the sensor picks up obstruction then the robot perform a 180 degree rotation and attempt the other direction. Assuming the robot turns to find an unobstructed direction the robot will advance one square turn parallel with the original path to determine if it can resume the original direction. If the pathway is clear it will advance two spaces forward and turn back in the direction of the where the vacated original path is to attempt to reacquire the original planed path. See Figure 4.6 for a representation of this avoidance maneuver.

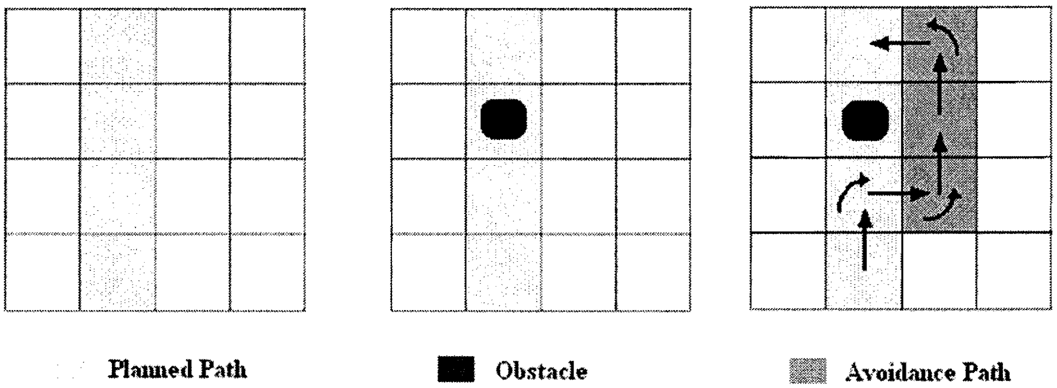


Figure 4.6: Representation of obstacle avoidance maneuver

The resultant path required to avoid the obstacle will be then set as the planned path for future operation. Note that the original planned path is still held in memory until manually changed because each time the robot reaches the point which it originally sensed the obstruction it will reevaluate and determine if the obstruction still exists. Only if it does indeed still block the original planned path will the robot use the newly generated avoidance path.

4.2 DARPA

4.2.1 Introduction

The DARPA robotics platform is a very advanced robot which has already been acquired for a previous application in 1996. As per the FORs which require the sensor suite have the capability of universal application on a variety of robotics platforms the DARPA is to be used, as time allows, to demonstrate this ability. However due to the age of the robot if will be necessary to determine the current workable status of the platform before integration consideration may be approached. The DARPA platform is pictured in Figure 4.7.

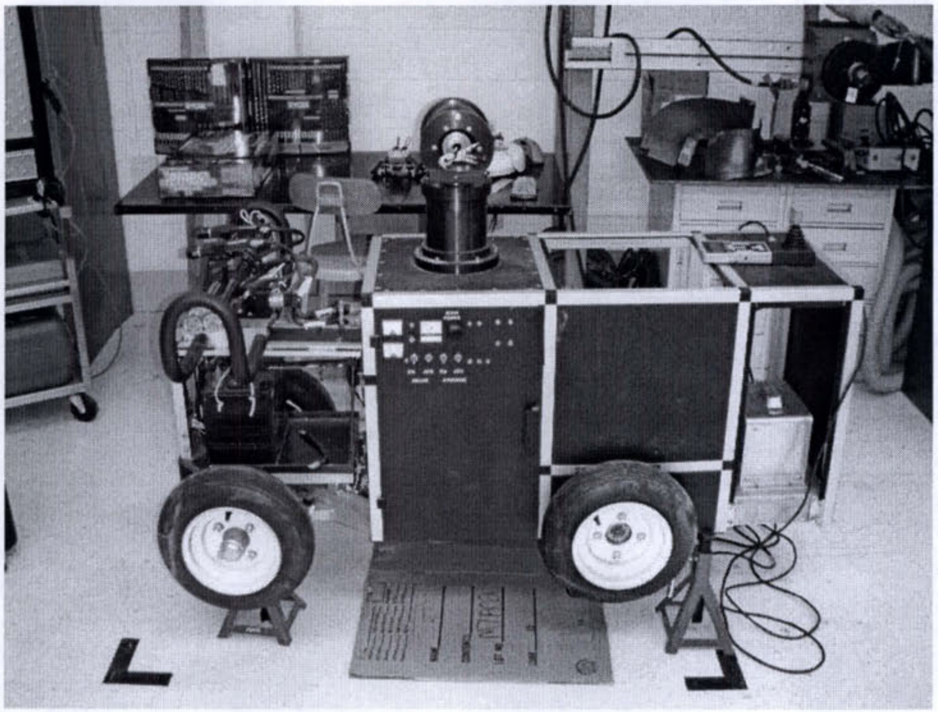


Figure 4.7: DARPA Robotics Platform

4.2.2 Current Status

Having been unfruitful in the search for the detailed operational manual for the DARPA platform most of the current technical data for this platform has been determined through observation. The DARPA uses the same wheel encoder technology as the Create however one of these encoders is loose and must be welded back to the wheel frame. The bottom of the transmission shows signs of an oil leak which based on the duration of time that the platform sat could mean that the transmission is dry. This must be determined before the robot is turned on due to the possibility of grinding the gears. Additionally the age of the batteries requires that they be replaced. The batteries are a low amp extended hour battery design for marine applications. There is no reason to not replace the batteries with an exact replacement.

The one piece of documentation discovered was with the robot itself and entailed the unpacking instructions which luckily included a majority of the pendant controls. These controls were provided in this unpacking document because the robot was shipped hot so that the DARPA platform could be driven out of the shipping crate. The Pendant is a wired remote operator for the platform. It is capable of controlling most if not all of the locomotive functions as well as the platforms impressive pan and tilt mount. This is an important find because it serves to control the majority of the functions that would be required of the platform for this demonstration, thus providing the most probable way of integrating the sensor suite with this DARPA platform. While the onboard CPU would provide a power asset to the autonomous operation of the unit not having the operational manuals have rendered its processing power all but unreachable. This is why the possible duplication of the inputs from the pendant control in order to provide the automation of the robot is the most likely route for the successful use of this platform. The pendant hookup as shown in Figure 4.8, is well suited to the possibility of cloning the pendant commands and driving by signal provided by a remote computer if appropriate wire jumpers are connected. The main loss of this operational solution is the use of the wheel encoders, it will be necessary to determine a substitute for this data in order to use the same operating control of the Create.

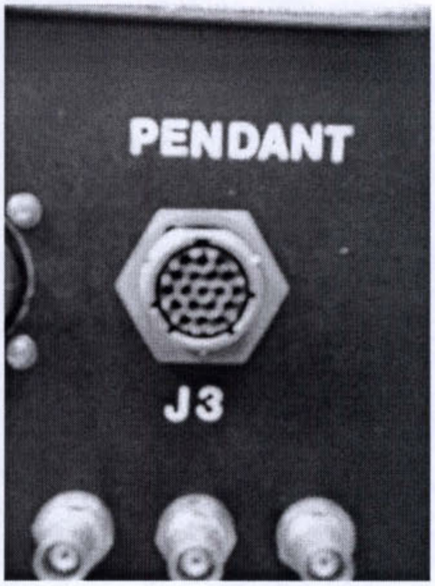


Figure 4.8: Pendant hookup on the DARPA Platform.

It must be clear that the use of the DARPA robotics platform is secondary to the use of Create and would be used only confirm the universality of the sensor suite package. Additionally, the entire use of the DARPA Platform will be conditional to the successful pendant operation upon acquisition of batteries and determination the drive train's mechanical soundness.

5. INTEGRATION

5.1 Introduction

The next phase of the design process is integration; following is an introduction to the hardware, software, and material requirements involved in incorporating the sensor suite to Create. Although introduced in Section 2 under system architecture, it is imperative to look at the system more technically.

5.1.1 System Diagram

In Figure 5.1 a system diagram can be seen detailing the sensor suite. This diagram presents the same operational functions as did Figure 2.1, but with detail concerning the specific wiring, inputs, and outputs of each sensor. This diagram is subject to iteration, but this system diagram is the guiding foundation for the hardware integration of the sensor suite.

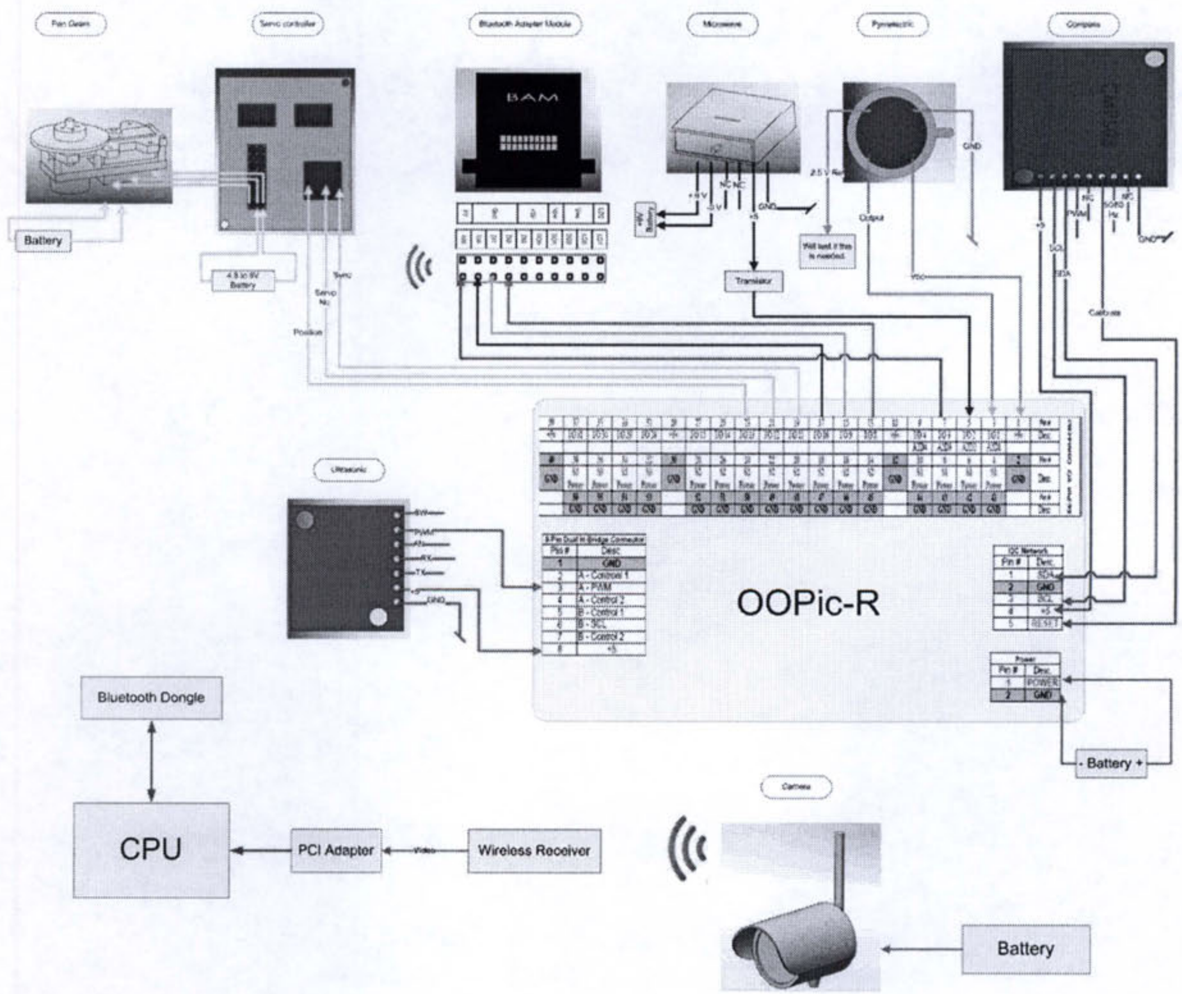


Figure 5.1: System Diagram

5.2.1 OOPic

OOPic is an acronym for Object-Oriented PIC. It is essentially a PICmicro by Microchip Technology that uses an objected oriented approach to control hardware. The concept behind object oriented approach is to use preprogrammed multitasking "Objects" from a library to do all of the work interacting with the hardware. Scripts written in Basic, C, or Java syntax styles control the Objects. The OOPic also has a "Virtual Circuits" capability, which is the software equivalent of an electronic circuit connecting Objects together in various ways. This feature allows Objects to pass data to each other in the background so that a script does not have to.

The OOPic that was chosen was the OOPic-R which has several I/O pins and within the budget. It has a 56-Pin I/O connector, four of which are analog. This is more then enough to serve our needs. In addition, the OOPic-R has an 8-Pin Dual H-Connector which provides two pulse width modulation (PMW) lines that will be used for the ultrasonic sensor. The principle of pulse-width modulation uses a square wave whose duty cycle is modulated which results in the variation of the average value of the waveform. The OOPic-R also has the Inter-Integrated Circuit (I2C) operating mode that will be used for the compass. I2C is a multi-master serial computer bus by Phillips and there are two I2C network built into the OOPic-R hardware that runs at 19,200 bps. From Figure 5.2, the lines consist of two open drains, Serial Data (SDA) and Serial Clock (SCL), Ground (Gnd), Reset (Rst), and +5 Volts (+5V).

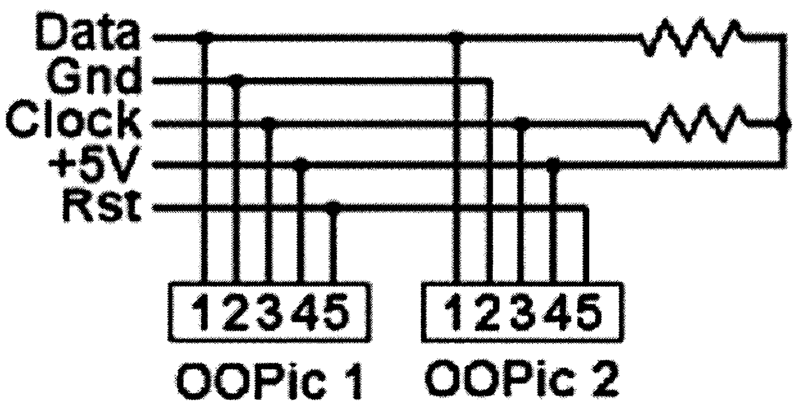


Figure 5.2: I2C Network

The PIR sensor, microwave sensor, ultrasonic sensor, and compass will be connected to the OOPic. There are three digital I/O line from the PIR, microwave, and ultrasonic. The compass requires an analog I/O lines. As of now, the microwave motion sensor will not be activated until the pyroelectric detects a change in temperature. However, it might be more beneficial to have both sensors "active" at all the times instead of having the pyroelectric trigger the microwave. Regardless, experiments will be required to determine the accuracy of either sensor detecting a "human" while on the move. All data acquired from the sensor will go through the OOPic first and then be sent to the host PC using wireless transmission for analysis.

5.2.2 Pan System

While the Create is moving from checkpoint to checkpoint, it will be limited to translational motion. As mentioned before in section 3.1.2, the pyroelectric sensor will need to pan so that a wider area can be covered without making the Create rotate. Since the live video will also need to pan, the pyroelectric sensor will be placed on the video for simplification of the system. The pan system consisting of a gearbox, servo, mount, servo controller, and power source will be required to accomplish this. Acquiring this system is within out budget, but will depend on how much power is needed to pan the camera and pyroelectric. The weight of the pan system will need to be minimized so as to lower the load on the gears and servo train. The servo controller board allows for the control servos using the OOPic-R by removing the included serial cable and replacing it with a signal wire from your microcontroller board. The servo controller board requires three data bytes: sync byte, motor/servo number, and motor position.

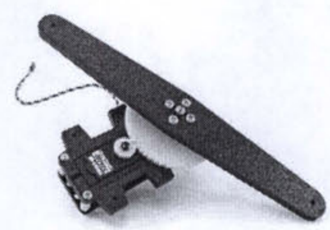


Figure 5.3: Servo Power Gearbox kit.

5.3 Material Requirements

Notice Figures 5.4-5.8, these figures display the two concepts for the construction of the sensor suite. There are two primary designs, one with a cylindrical frame and a second with a square frame. Both designs are satisfactory to the demands of the project, and each has a distinct advantage. The cylindrical design, version 2, appears to have a more finished look. The square design, version 1, will be easier to manufacture. Both designs are presented for completeness; although it is likely version 1 will be the final design concept due to its manufacturability. With that stated, considering the construction of the sensor suite chassis is just as important as the sensors themselves. Since there is a payload limit on the Create platform and the PIR cone is sensitive to extraneous data, due consideration to the manufacturing materials is considered.

The majority of the chassis is expected to be created out of Lexan and is designed to have easy access to the hardware. The clear construction and ease of access allows for simple troubleshooting during experimentation. Possibly for the final production, a finished cover can be used to conceal the interior for demonstrations. Threaded steel rods can be used as a frame since the Create has threaded bolt holes where it can be mounted without having to modify the platform.

A template was provided by Acroname for the construction of the PIR cone. However, the cone provided is constructed out of paper and it has been seen that this material is unacceptable due to the fragile nature of the paper construction. For reliability, the cone needs to be rigid to keep the Fresnel lens and PIR parallel. ABS plastic will probably be used to construct the PIR cone. Other possibilities suggested by Acroname for material are sheet metal, ceramics, and glass. Currently, most PIR cones are constructed out of ceramics, but ABS plastic will probably be the easier and cheaper material for this construction.

5.4 SolidWorks

All sensors and hardware will have a SolidWorks model. While not all models need to be an "exact" replica of the actual item, all models are required to have the interfacing dimension precise to 1/16 in. Below are the SolidWorks models and assemblies representative of the sensor suite concept.

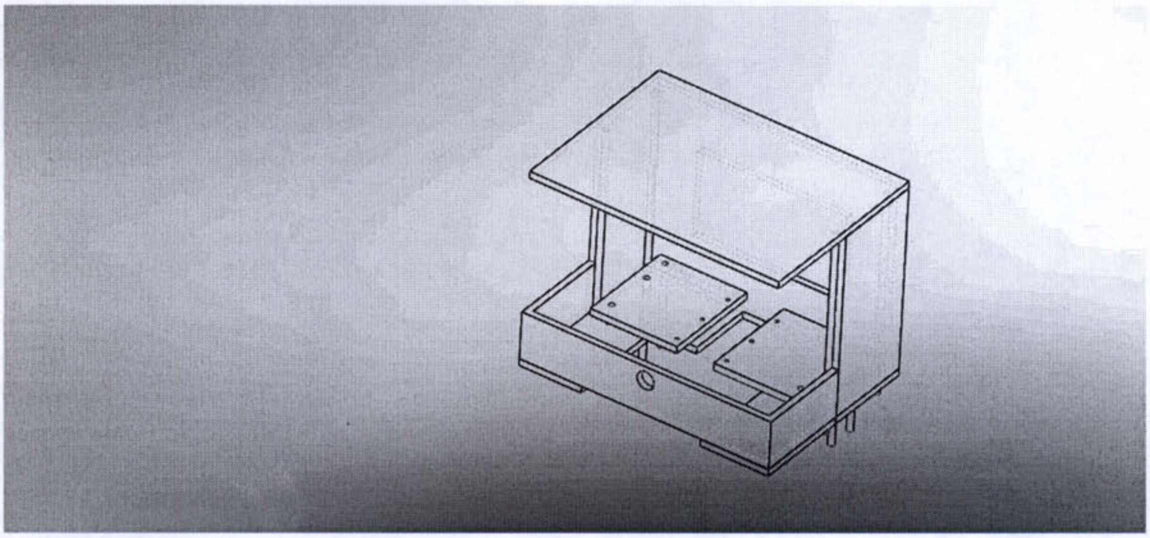


Figure 5.4: SolidWorks drawing of sensor suite chassis version 1.

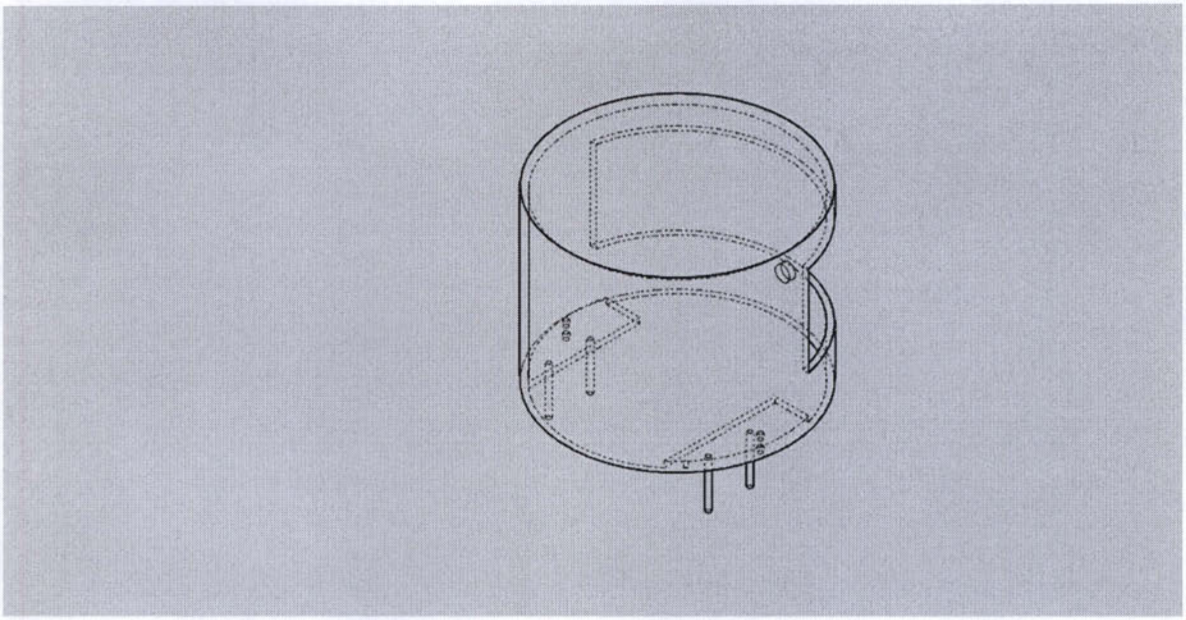


Figure 5.5: SolidWorks drawing of chassis version 2

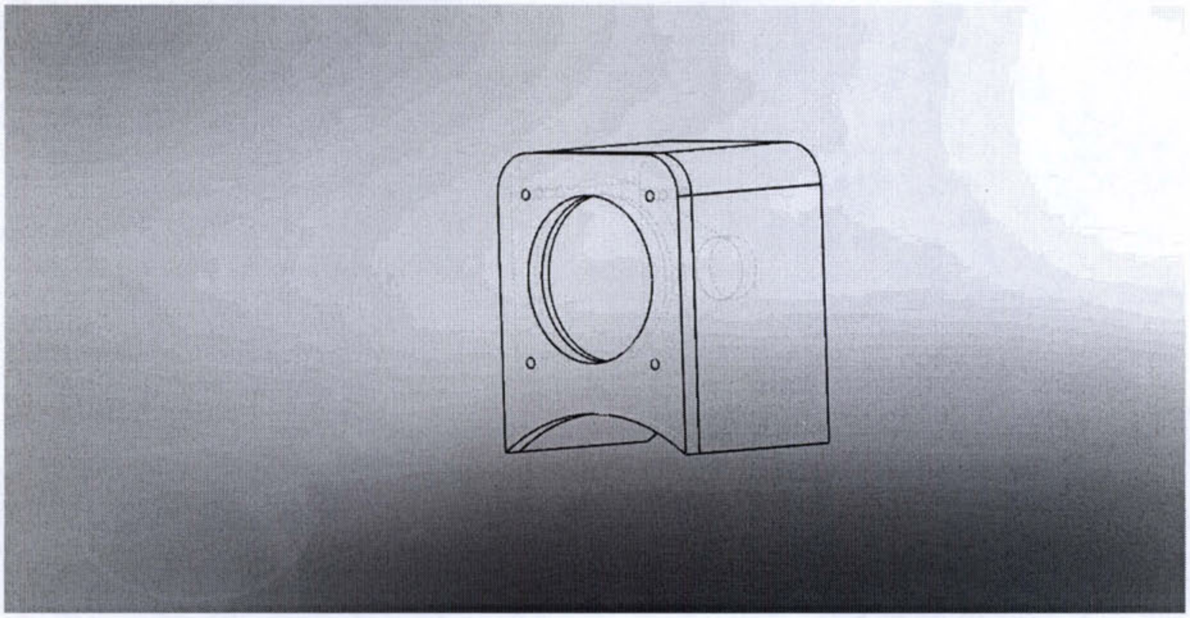


Figure 5.6: Model of Pyroelectric Cone.

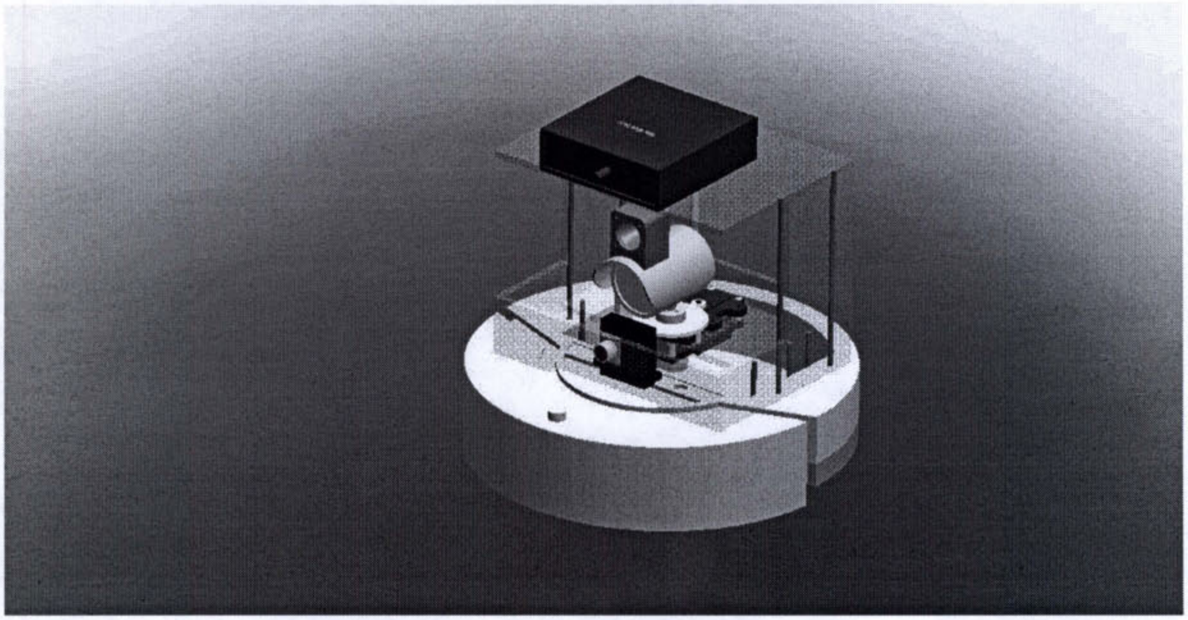


Figure 5.7: Create with Chassis and Sensors (Version 1)

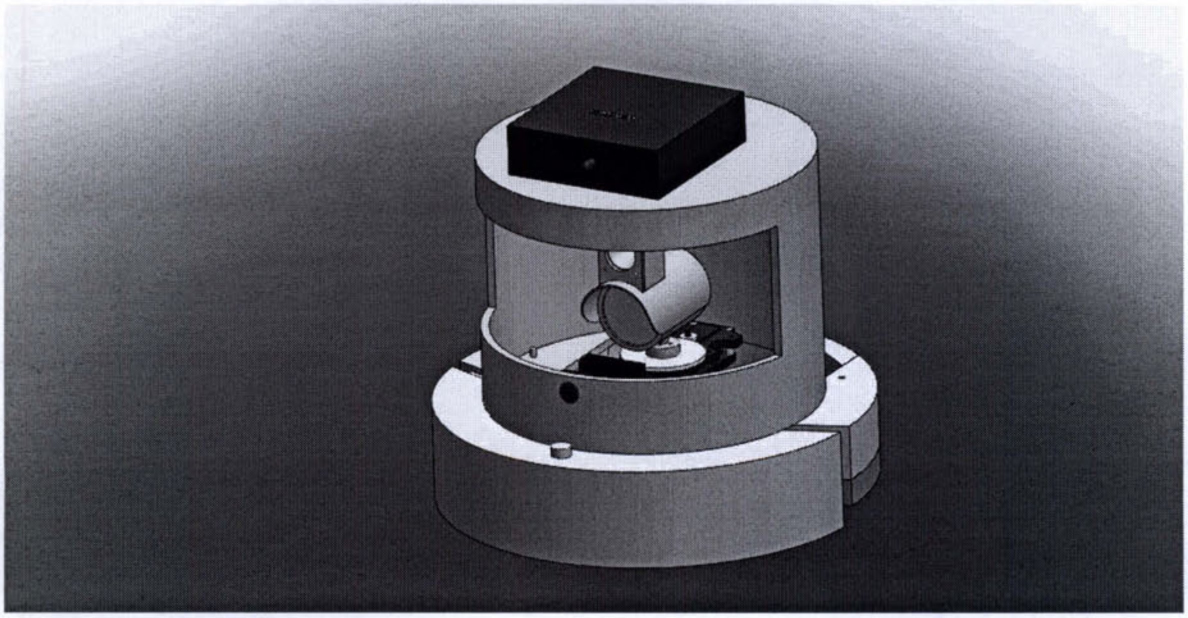


Figure 5.8: Create with Chassis and Sensors (Version 2)

6. Experimentation

6.1 Overview

The following discusses the completed experiment conducted on the pyroelectric sensor as well as the methodology for the remaining scheduled component and system tests.

6.2 Component Testing: Pyroelectric

Although some general specifications are available for the pyroelectric detector we are using, we wanted to perform diagnostics of our own. This would ensure that the specifications we received were accurate for our specific sensor as well as allow us to foresee possible obstacles to its implementation so they could be addressed early on.

The first pyroelectric detector experiment was performed on the afternoon of Sunday, November 18, 2007 in Dougherty 203.

Materials

The following materials were used in the experiment:
-pyroelectric detector (attached to a paper cone holding Fresnel lens)
-roll of wire
-solderless breadboard
-voltmeter
-9V battery
-roll of duct tape
-roll of twine
-lighter
-stopwatch
-unheated object to serve as shutter

Setup

In order to prepare the sensor for the tests, we cut off three pieces of wire of approximately equal lengths and then stripped the ends. We then applied each wire to one of three pins: the input voltage pin, the signal pin, or the ground pin. The wires to the input voltage and the ground were then connected to the solderless breadboard. Similarly, two other pieces of wire were cut and their ends stripped so they could be duct taped to the two poles of the 9V battery. The positive wire from the 9V battery was connected to the input voltage wire of the sensor. The negative wire from the 9V battery was connected to the ground wire of the sensor. Finally, a voltmeter was connected across the ground and signal wires of the pyroelectric sensor. (See figure below.)

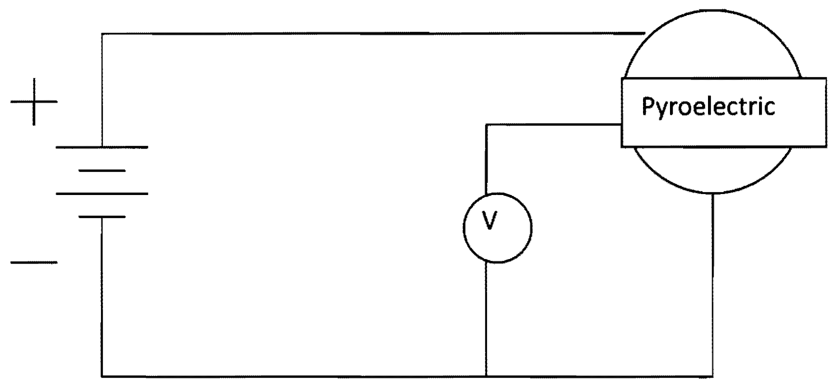


Figure 6.1: Circuit diagram of the pyroelectric setup.

The pyroelectric sensor, breadboard, and battery were then placed on the floor of the room with the pyroelectric sensor mounted on top of the cardboard box that the iCreate robot was shipped in. This was to simulate the height at which the pyroelectric sensor would be on MoHSeR. The sensor faced along one of the lines of tile on the floor for a consistent, albeit flawed, way of estimating the field of view of the sensor.

A long piece of twine that could stretch the entire length of the room was duct-taped to the floor directly in front of the sensor setup for use in angle experiments.

When simulating low-light conditions, the shades were pulled down and the lights were turned off.

Procedure

The following procedure was followed during the experiment:

1. Turn sensor on, and see if any change in voltage (ΔV) can be detected. (In other words, check for noise.)
2. Turn on sensor with human standing directly in front of it at a known distance. Does the sensor detect the human? If so, attempt to measure the peak voltage spike (ΔV). Also, note

how long the signal takes to decay (if possible, time this). Repeat at different distances to see whether the signal varies significantly with distance.

3. Have human stand still at a known distance directly in line with the sensor. Let signal go to steady state. Cover sensor with shutter and then uncover it. Does this affect the signal? If so, is there a shutter frequency at which the signal stays approximately constant?
4. As close to a direct line to the sensor as possible, have human choose arbitrary distance to stand away from the sensor and turn on the sensor. Note ΔV. If human is detected, have human move farther away from the sensor along the direct line and retest. If human is not detected, have human move closer along the direct line and retest. Repeat until sensor threshold seems evident. Measure this distance and record. Repeat for two different people to see if this results varies for different people.
5. As close to a direct line to the sensor as possible, have human choose an arbitrary distance to stand away from the sensor and turn on the sensor. Now have the person take some twine and pull it taut so that one end is at the sensor and the other is in control of the sensor. Have human swing along an arc until the sensor cannot detect him/her. Mark the entrance and exit locations for the radius and repeat. Once all the measurements are made, calculate a best fit line for the coordinates found and determine the angle of the sensor's field of vision.
6. Introduce non-human heat source (lighter). Can the sensor detect it? How does this ΔV differ from that of a human's (if at all)?
7. Simulate low-light conditions: turn out the lights and lower the shades. How does this affect the sensor's readings?

Results

Once the pyroelectric detector setup was complete, we turned on the voltmeter at a resolution of 1×10^{-3}V, only to find that the noise rendered that resolution almost useless. This might not be the case if we were somehow graphing the data realtime, but for the purposes of our experiment we needed to read from a volt meter. At a resolution of 1V, the steady state voltage difference of the pyroelectric appeared to be 2.5V, with noise occasionally bumping it around anywhere in the area of 2.3V to 2.6V. We thus defined those limits as our limits of detection – if the voltage were to exceed 2.6V or to go under 2.3V, we would consider the human detected.

We noticed that turning the sensor on was not sufficient for it to detect a human immediately. When the sensor was not supplied power, the voltmeter read 0V. When power was applied, the voltmeter jumped up to 8 or 9V (the voltage of the power source), and then died down asymptotically to 2.5V or thereabouts. Thus, we could not simply apply power with a person standing in the range of the sensor – the results would be inconclusive. Instead, we found that entering and exiting the field of view of the sensor would register as a detection. This wasn't disturbing since the sensor will be rotating back and forth on a pan-and-tilt mechanism, so even still objects will constantly be entering and exiting the sensor's field of view.

The decay rate of the signal for the pyroelectric detector was fairly rapid, which introduced a measurement problem. Since we were using a voltmeter to measure the signal, we had extreme difficulty measuring peak values of the sensor's signal voltage (we could usually only tell the peak ones place with any accuracy). Thus some of the planned measurements in the procedure were unobtainable. However, we intend to perform this set of experiments again when our capabilities are such that we can graph the data over time and see where the peak was.

We did discover that the closer a human walks in front of the sensor, the higher the voltage response. At very close range (within 4 feet), the sensor spiked at various voltages from about 3V to 4V (averaging to 3.5V, or 1V above the reference voltage). At farther ranges (4+ feet), the sensor still spiked, but it was more on the magnitude of 0.5V above reference - it was definitely above the voltage range of noise, but only slightly so.

Shutter tests were ineffective. Using a shutter seemed to have no effect whatsoever on the signal of the sensor, as no frequency of covering or uncovering the shutter seemed to cause any noticeable voltage change.

For this setup, we were not able to find the upper limit of the sensor's detection range. Subjects were detected all the way to the other end of the room, a distance of approximately 20 feet, so we know that the sensor has at least a range of 20 feet. For next time, we will also move the testing setup to the hall so as to find conclusively the range at which the motion of a human cannot be distinguished from noise. This should be more dependable as well if we can obtain software capabilities to see the response of the sensor.

In order to calculate the angle of the sensor's field of vision, twine was tied to a subject's shoe, and the subject walked in an arc about the sensor, careful to keep the twine taut. When the signal peaked on the sensor, it indicated that the human had entered its field of vision. Once the limits were established, a piece of duct tape was placed on the floor to indicate the location of the edge of vision. This was repeated for different lengths of twine. Then, x and y coordinates were measured for each piece of tape: the x measured the distance along the line of tiles that the duct tape piece laid and the y measured the distance away from the line of tiles of the duct tape piece. These were put in Excel and lines of best fit were found. See figure below.

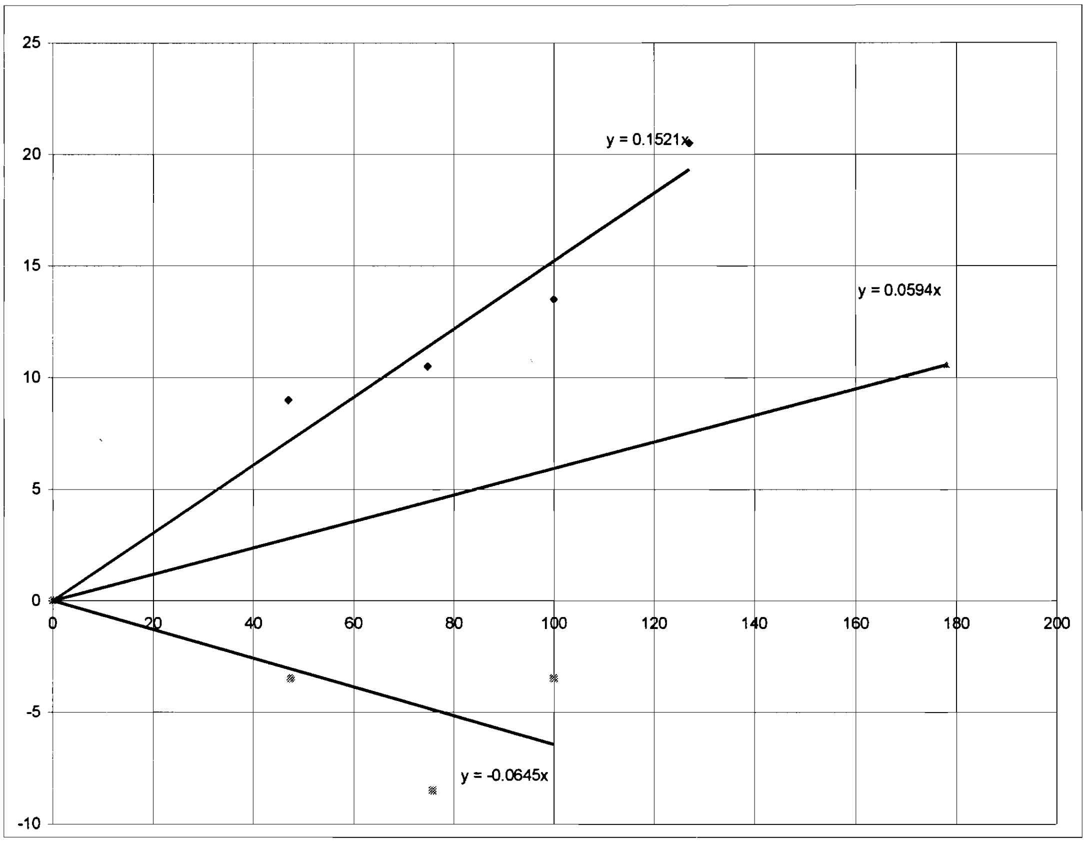


Figure 6.2: Angle Data from Pyroelectric Detector

As can be seen above, the overall best fit line (which represents the center line) had an equation of y1=0.0594x. The best fit line for the readings to the left of the center had an equation of y2=0.1521x. For the readings to the right of the center, the equation was

y3=-0.0645x. From these three equations, an angle can be calculated for the sensor's field of vision.

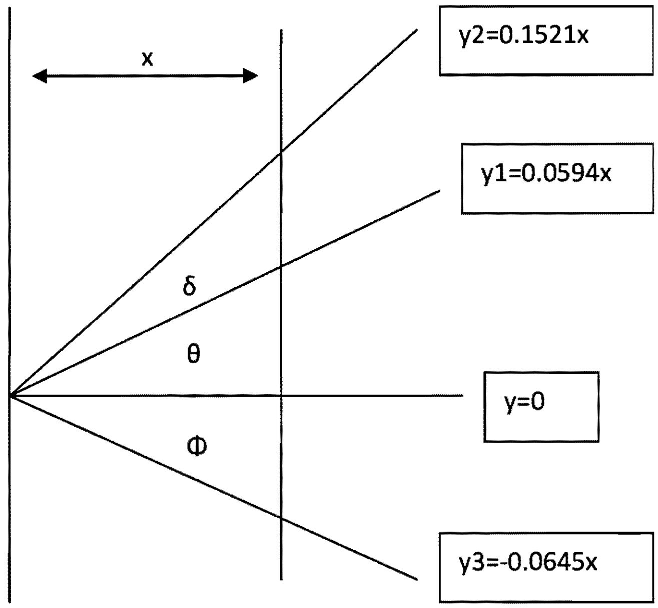


Figure 6.3: Calculating the angle for the sensor's field of vision

Let θ be the angle between the center-line (represented by y1) and the x-axis (represented by y=0). Let δ be the angle between the center-line and the outer edge of sensor readings to the left (represented by y2). Let Φ be the angle between the x-axis and the outer edge of sensor readings to the right (represented by y3). Then from figure ## above:

$$(\theta+\delta)=\tan^{-1}\left(\frac{y_1+y_2}{x}\right)=\tan^{-1}(0.0594+0.1521)=11.9421^{\circ}$$

$$\Phi=\tan^{-1}\left(\frac{y_3}{x}\right)=\tan^{-1}(-0.0645)=3.6905^{\circ}$$

The estimated angle of the sensor's field of vision from the graph would then just be the addition of these two angles, or 15.6 degrees.

As is evident from the figure, the center line was at an angle, not quite horizontal. We believed this could indicate that the paper cone holding the lens could be slightly off, or the lens itself could be being held at a slight angle.

When a lighter was lit a few inches away from the sensor, it was easily detected. Indeed, the sensor spiked higher than it had spiked before – somewhere between 5 and 6V (or 3-4V above reference voltage). This could be valuable to know if a human never spikes the sensor this high, as this could be filtered out with software.

Under low-light conditions, however, it was determined that all readings were magnified in terms of signal. Humans at a medium range in low-light conditions generated readings as high as they would have in short range in the light. We believe this is because in the dark the temperature of the surroundings is cooler, while the human is still the same temperature. Thus when a human enters the field of vision of the sensor, a greater change in average temperature of the sensor's field of vision is detected, leading to higher readings. This means that MoHSeR has the capability to detect its surroundings better in conditions that a human would have more difficulty, which fits very well in with the purpose of the robot.

As far as repeatability is concerned, the pyroelectric detector appeared to detect people consistently within close-range, although it was less reliable at greater distances. It probably detected their heat presences every time, but only at times was the detection distinguishable from the noise.

Conclusions

Perhaps the most striking thing learned from this experiment was how badly we needed a computer interface to analyze the data from the pyroelectric sensor. If we could see the voltage of the sensor plotted against time, we could vastly increase the accuracy of our measurements. This is very important for us to do, since we need to program the OOPic to parse the pyroelectric data in the most intelligent manner possible. Thus, we will be performing another pyroelectric sensor test later with these capabilities if possible.

Another aspect that may help in the future test (as well as on the robot) is the incorporation of an integrated circuit called the MAX232A. We believe the inclusion of this chip will help stabilize the pyroelectric sensor's reference voltage at 2.5V. If this does happen to be the case, it will greatly boost the sensor's signal-to-noise ratio, since the noise will be lowered significantly. (If it isn't the case, it's little loss, since we're receiving free samples of the chip through the MAXIM company.)

We did receive a fair estimate of the range of angles the sensor could detect. However, we also noticed that the sensor was not completely centered. While it's possible that this could partially be due to an inaccurate setup, the degree to which the sensor is off-center suggests that the paper cone may not be holding the lens in the right position. Thus we are actively pursuing the creation of a new cone made of a sturdier material in hopes of improving the accuracy of the sensor's readings. The next test will be performed with this new lens holder installed on Sunday, January 13, 2008 in Dougherty 203.

6.3 Component Testing: Microwave Motion Detector

Again, although a good amount of technical data was available on the microwave motion detector, we wanted to verify this data for our specific model. We also wanted to determine the answers to some unanswered questions and try to find out if any immediate problems arise that could make its implementation simpler.

This test will be performed on the afternoon of Sunday, January 20, 2008.

Materials

The following materials will be used in the experiment:
-microwave motion detector
-roll of wire
-solderless breadboard
-9V battery
-roll of duct tape
-roll of twine
-transistor
-resistor
-LED

Setup

In order to set up the microwave motion detector, several pieces of wire must be cut and stripped. Two will be used to connect the 9V battery to the motion detector power input and ground pins. One will be used to connect the signal pin of the motion detector to the solderless breadboard. There a circuit will be set up with a resistor, transistor, and LED so that a signal from the motion detector will light the LED.

The microwave setup will be placed on the box that the Create came in to simulate the height of MoHSeR. A long piece of twine will be taped to the floor directly in front of it.

Procedure

1. Turn on the motion detector and watch for any erroneous signals. Do high sensitivities produce noise?
2. Walk normally in front of the motion detector the lowest sensitivity setting. What is the largest distance at which the motion detector can detect motion at this setting and pace? Try again at the highest sensitivity, as well as some sensitivity in the middle.
3. Now walk slowly in front of the motion detector at the lowest sensitivity setting. What is the largest distance at which the motion detector can detect motion at this setting and pace? Try again at the highest sensitivity, as well as some sensitivity in the middle.
4. Is there some speed that one can cross in front of the motion detector at any distance without being seen? Try this on low sensitivity and high sensitivity.

5. What is the range of angles that the motion detector can see? Set the motion detector to low sensitivity and swing in an arc about the sensor to find the extremities. Mark these on the floor with duct tape, and later find lines of best-fit for the data. Repeat this experiment for high sensitivity to see if the angle varies.
6. Does the motion detector perform differently in low-light conditions than in normal-light conditions? If so, how?

6.4 Component Testing: Sensors (Odometer, Beacons, Transmitter, Compass, Sonar)

In the interest of time, we are planning to perform tests for all of the location sensors at the same time. Individual tests of these parts are less intensive than other experiments to be performed, so this will probably not take much more time than one of the other experiments.

This experiment is scheduled to be performed on the afternoon of Sunday, January 27, 2008 in the hallways of Dougherty Hall.

Materials

-Create robot
-Infrared beacons (with indicator lights)
-Transmitter set
-Electronic compass
-Roll of duct-tape
-Laptop
-Protractor
-Roll of twine
-Sonar sensor
-Stopwatch

Setup

The Create will need to be fully charged, and appropriate programs will need to be uploaded to the Create.

Procedure

The first tests to be performed will evaluate the odometer function of the Create robot.

1. Tell the Create to go forward 1 meter and stop. Measure the distance traveled. Subtract the difference to find the error. Repeat this twice more and average the results.

2. Tell the Create to go forward 10 cm, stop, and repeat those two steps ten times (for a total distance traveled of 1m). Measure the distance actually traveled, subtract the difference, repeat twice more and average the results. Compare this to the results from Test 1.
3. Tell the Create to go forward 1 cm 100 times. Repeat the process from Tests 1 and 2.
4. Try going forward 1 meter at a speed faster than that used for Tests 1-3. Calculate the error from going forward at this speed. Try again with a slower speed. How does speed affect the error in each case?
5. Repeat Test 1, except with a change in ground surface (e.g., tile and carpet) in the middle of the 1 m traversed. Does this increase the error? If so, how drastically?

The next series of tests will examine the properties of the infrared beacons.

6. Place an infrared beacon against a wall and turn it on. Tell the Create to go forward until it encounters an infrared signal, pause until told to continue, and then go forward until it loses the infrared signal.
7. Orient the Create so that it crosses the line of sight of the infrared beacon at a perpendicular angle. Mark the spots the Create stops at with duct tape, and then repeat at different distances. Analyze this data to determine the range of angles at which the infrared beacon broadcasts.
8. Now tell the Create to go forward until it loses the infrared signal. Place its rear directly in front of the beacon and then activate the program. Mark the spots at which the Create loses signal, and then repeat twice more. The average distance from the beacon represents the range of the beacon.
9. What is the battery life of the infrared beacon? Make sure the indicator lights on the infrared beacon are set up and then leave the beacon on until the lights go off. Record the time required for the batteries to die. (Note: since indicator lights are not normally used on the beacons, the actual battery life will be slightly longer than it will be in test conditions. This will just be to get a general idea.)

Next up in the testing order is the Bluetooth transmitter. The Create transmitter/receiver will be connected to the Create robot's serial port, and the USB transmitter/receiver will be connected to a laptop computer.

10. Verify Bluetooth connection between the laptop computer and the Create robot.
11. Tell Create robot to send data, wait for receipt of data, then go forward a very small distance and repeat the process. This should give a reasonable estimate of the range of the transmission. Mark with duct tape the outer edge of the Create's reception, and then repeat twice more. Measure each distance and average the results.
12. Tell Create to send data and wait for receipt of the data. Upon receipt, the Create should spin in place for one revolution and repeat. While the Create does this, move the laptop around the hall, experimenting with different orientations of the antenna. Are some orientations more favorable to reception than others? Also, see if obstacles or walls between transmitter and receiver reduce the strength of the signal, and if so, by how much.

13. Tell Create to wait for receipt of a signal from the laptop and then move in a straight line for one meter. Use a stopwatch to see how long it takes from sending the signal to the robot moving. See if this varies noticeably for different distances.

The next test will be covering the electronic compass device. For the purposes of this test, the compass will be connected only to the OOPic and a power source. An LED will be connected to the OOPic. The OOPic will run a simple program that will light the LED if the compass tells it that it is facing due north.

14. Put the compass on the floor and rotate it until the OOPic indicates that it is facing due north. Mark this spot with duct tape.
15. Without moving the compass away from its original location on the floor, rotate the compass several times and then find due north once more. Did due north change position? If so, by how much?
16. Using a map of the University of Tennessee Campus, verify that the compass is indeed facing due north. It is more important that the compass reliably point in one direction than it is that the compass point in the right direction, but it is still important to know whether the compass is in fact in error, even if consistently.

The final test involves the sonar sensor. The sonar sensor will be attached to a power source and voltmeter in a similar manner to the experiments with the pyroelectric detector – the voltmeter will measure the difference in voltage between ground and the signal.

17. Verify that the sonar detector can detect something 12" or less away, and find the associated voltage with that reading. Repeat several times and see how the voltage fluctuates with proximity.

6.5 Sensors – Camera

Component Testing: Camera

The video camera is perhaps the most complicated onboard sensor used in MoHSeR. As such, there are many questions to be answered about it. Experimentation should provide a good starting point for familiarizing ourselves with the camera's capabilities.

This experiment will be performed in the halls of Dougherty on Sunday, February 3, 2008.

Materials

-camera
-base computer station

Setup

The base computer station must be set up to accommodate the wireless kit, and the camera must be connected to its battery – otherwise, not much setup is required at all.

Procedure

1. Once the base computer station is set up, verify that the image from the video camera is being transmitted successfully to the base computer station.
2. Test the audio on the camera and verify that this audio is transmitted also, and syncs with the video.
3. Exit the base computer station and walk down the hall until the signal from the camera is lost on the base computer station. Repeat this several times until the extreme location of the camera signal is found. Measure this distance.
4. Test the audio's maximum range in the same way to see if the audio has a different limit than the video.
5. Test the video delay of the camera by signaling to the person on camera to do something and timing the delay between the action being performed in real life and on screen. Does this vary with distance?
6. Turn out the lights in the hallway and see how the camera performs under low light conditions. Can detail still be discerned?
7. Lightly shake the camera to see how well the camera reacts to vibration. Does the camera lose focus? If so, how quickly does focus return?
8. Test the battery life of the camera – how long does it take for the camera to stop sending a signal? Does this vary with the distance from the base computer station?

6.6 Platforms Test: iRobot Create

The platform is one of the most important parts of the robot, as without it MoHSeR is completely immobile. Thus it's important to know how well the platform will perform in various areas.

This experiment is scheduled to be performed on the afternoon of Thursday, February 7, 2008 in the hallways of Dougherty Hall.

Materials

-Create robot
-roll of duct tape
-stopwatch
-set of incremented weights
-charger
-tape measure

Setup

In preparation for this experiment, the Create robot will need to be programmed simply to go forward at full speed. It will also need to be fully charged.

Procedure

1. How long does the Create robot run on one charge without any load? Let it wander around until it stops moving. How long does it take to recharge fully?
2. What is the Create's top speed without load? Measure the time it takes to cross a given distance with a stopwatch, then repeat several times and average the results.
3. What is the Create's turn radius? Can it turn in place? If not, measure the diameter of the circle it can turn about and divide in half.
4. How much weight does it take for the Create to be unable to move? Apply known increments of weights to the top of the Create and see how many it takes to incapacitate the robot.
5. How much does weight affect the Create's speed? Repeat the top speed test with a load of three-quarters of the Create's max capacity, and again with a load of half the maximum capacity and one-quarter. Record the results.
6. How does load affect the Create's battery life? Again, use loads of one quarter, one half, and three-quarters of the Create's max capacity and let the Create run until it runs out of battery power.

6.7 Integration

Integration Testing

Once all of the components have been tested individually and coding has been written for MoHSeR, we need to verify that everything works together as expected. If there are no problems, then the robot will be easily cleared to participate in the demonstration. If problems arise, they need to be addressed and fixed before proceeding further.

The integration experiments will begin on Sunday, April 13, 2008 in Dougherty Hall.

Materials

-MoHSeR
-base computer station
-infrared beacons

Setup

Some of these tests will be conducted in Dougherty 203, in which case no setup is required. These are the tests that verify that there is no interference between different components of the robot. Other tests will be performed in the Dougherty hallway of the final demonstration. For these tests, several doors will need to be opened, a blueprint and the path-following and object detection program will need to be uploaded into MoHSeR. The base computer station will also need to be set up. There will also need to be relatively few people about, since we will be turning out the lights (which might startle those unfamiliar with our project). Infrared beacons will need to be placed at predetermined locations to provide a grid for MoHSeR to familiarize itself with its location.

Procedure

In Dougherty 203, turn on MoHSeR and all subsystems.

1. Verify that infrared beacons do not interfere with pyroelectric detector by bringing up a beacon in front of MoHSeR and seeing if it is detected.
2. Verify that panning motion does not trip microwave motion detector.
3. Verify that microwave motion detector is not continuously set off by Bluetooth signal.

If all of the above tests check out, the robot at least does not self-destruct and should be taken to the hallway of Dougherty.

4. Verify that MoHSeR can follow its path-following program without significant error due to miscalibration.
5. Verify that MoHSeR can adapt to unexpected objects in its path (e.g., doors), evade them, and return to its path-following program without significant calibration errors.
6. Have a human pace back and forth along MoHSeR's path. Does MoHSeR successfully detect the human? Can it send an accurate location marker back to the base computer station, including a picture of the perpetrator and reliability factor?

6.9 Final Demonstration

The purpose of the final demonstration is to verify that the robot is capable of meeting every one of its Functional Operational Requirements (FORs). It will also be a means of showing others what we have accomplished this year.

This test will be performed at night on the fourth floor of Dougherty on Thursday, May 08, 2008.

Materials

The following materials will be required for this experiment:

-MoHSeR

-intruders
-prizes
-base computer station

Setup

For this experiment, the fourth floor of Dougherty will need to be evacuated of all people except those involved with the experiment. MoHSeR will be placed at one end of the hallway, and one prize will be placed at the other end. Four predetermined doors will be opened to random degrees, thus protruding into the hallway to block the robot's route partially. One prize will be hidden in each open room. In another room, the base computer station will be set up, manned by a security guard who will be alerted if the robot detects anything unusual. Infrared beacons will be placed at set increments on the floor to give MoHSeR feedback on its current position. Once all of this is set up, the lights will be extinguished.

Procedure

MoHSeR will be activated and the intruder will enter from either doorway. MoHSeR will patrol its predefined route, which will include entering each open room and scanning for unusual activity. While MoHSeR patrols, the intruder will attempt to find and collect all five prizes undetected. The intruder will then try to exit the building with all five prizes and without being seen by MoHSeR.

MoHSeR will be monitored to answer the following questions:

1. Does MoHSeR stop when a heat source is detected? Does it send back a picture and reliability factor to the base computer station?
2. How well does MoHSeR navigate its programmed path (when possible)?
3. How well does MoHSeR react to unexpected obstacles? How long does it take for MoHSeR to get back on track?
4. How accurate is the position data that is relayed to the base computer station?
5. Does MoHSeR detect the intruder before the intruder escapes?

Note that these questions roughly correspond to the FORs as established earlier in the report.

This test will be repeated once for every team member so that everyone has a chance to be both an intruder and a security guard.

7. PROJECT STATUS

7.1 Executive Summary

MoHSeR is currently poised to make the transition from the research and design phase of the engineering process to the acquisition, implementation, and experimentation phases. As has been detailed above, this project was aggressive from its inception due to resource limitations - primarily time. With that stated, a consistent effort has been made to plan the next phases of the project. Included below is a work break down structure, budget review, and project timeline. Accordingly, the project management techniques will be discussed along with the proposed end-of-project deliverables.

7.1 Progress Report

The first semester provided a steep learning curve and was a success. As seen from the report above, we were able to define parameters and limitations for MoHSeR. Additionally a set of milestones were established to determine the progress and success of the project. From this, we were able to develop a system of sensors that met these challenges. Finally, these sensors are designed to be compatible with each other for implementation onto the platforms.

This semester can be termed a success since all but one of the milestones was met for this portion of the project. The one milestone not met is the acquisition of all hardware before winter recess, because of this we have compensated by planning to acquire the hardware during winter recess. See Table 7.1 for a comparison of the milestones met for this project.

Table 7.1: The Milestones for the Research and Design Phases

Milestones	Status
Idea Generation and Project Conception	A strong desire to apply a pyroelectric sensor to a mobile platform was expressed. Idea generation stemmed from this desire to develop a mobile human seeking platform. This milestone was reached 9/5/07
Define Project Requirements	An independent Functional and Operational Document was generated and submitted for review on 9/28/07. This document is transient due to design changes and improvements.

Acquire Funding	A grant of $2,300 was awarded 10/25/07. An account has been created. The budget was developed according to the grant value.
Research Current Sensor Applications	Presentations, interviews, and design reviews were conducted to find the most applicable sensors. These sensors were finalized in a design review held: 11/21/07
Develop Sensor Suite Design	Design reviews and 3D drawings generated to proof mechanical and electrical concepts.
Develop MoHSeR System	Much work was afforded to the development of a system diagram that considered such components as weight, input, output, and operational limitations. This milestone was first accomplished 11/30/07. This document is living, or subject to revisions, due to design changes and improvements.
Acquire Hardware	The pyroelectric sensor was the only hardware acquired. All other sensors will be ordered shortly (over winter recess).
Develop Strategy for Assembly and Demonstration	Business Review Meetings, Timelines, and Milestones have been discussed and published to provide an outline for implementation.

7.2 Project Timeline

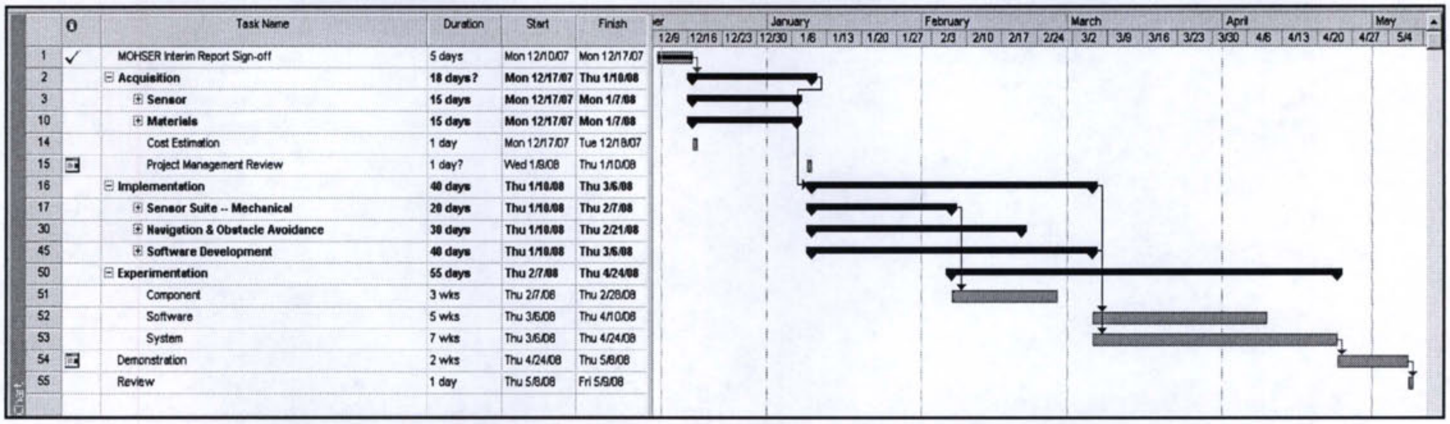


Figure 7.1: The planned timeline for the next phases of the MoHSeR project.

The timeline for the Acquisition, Implementation, and Experimentation Phases can be seen above in Figure 7.1. The current structure calls for the sensors and chassis materials to be acquired by January 1, 2008. This will provide an efficient use of time for winter recess, allowing time for ordering and delivery. We intend to have the mechanical portion of the sensor suite finished no later the four weeks into the semester – or by February 7, 2008. With the mechanical milestone complete, it will allow all resources to be focused on developing the navigation and software systems. These systems will be developed in parallel with the mechanical chassis, but are also given more development time at about seven weeks. Similarly, component testing can begin as soon as the sensors are acquired (See Section 6.3). Testing will continue throughout the entire semester; the project is designed to test progressive experiments and milestones until the project is completed. The final test is a demonstration which is scheduled to be completed on May 8, 2008.

Table 7.2 Future Milestones

Milestone	Date
Hardware Acquisition Complete	1/9/08
Sensor Suite Complete (Mechanical)	2/7/08
Navigation Integration Complete	2/25/08
Component Testing Complete	3/1/08
MoHSeR Software Fully Developed	4/9/08
MoHSeR Integration Testing Complete	5/2/08
MoHSeR Demonstration	5/8/08

7.3 Budget

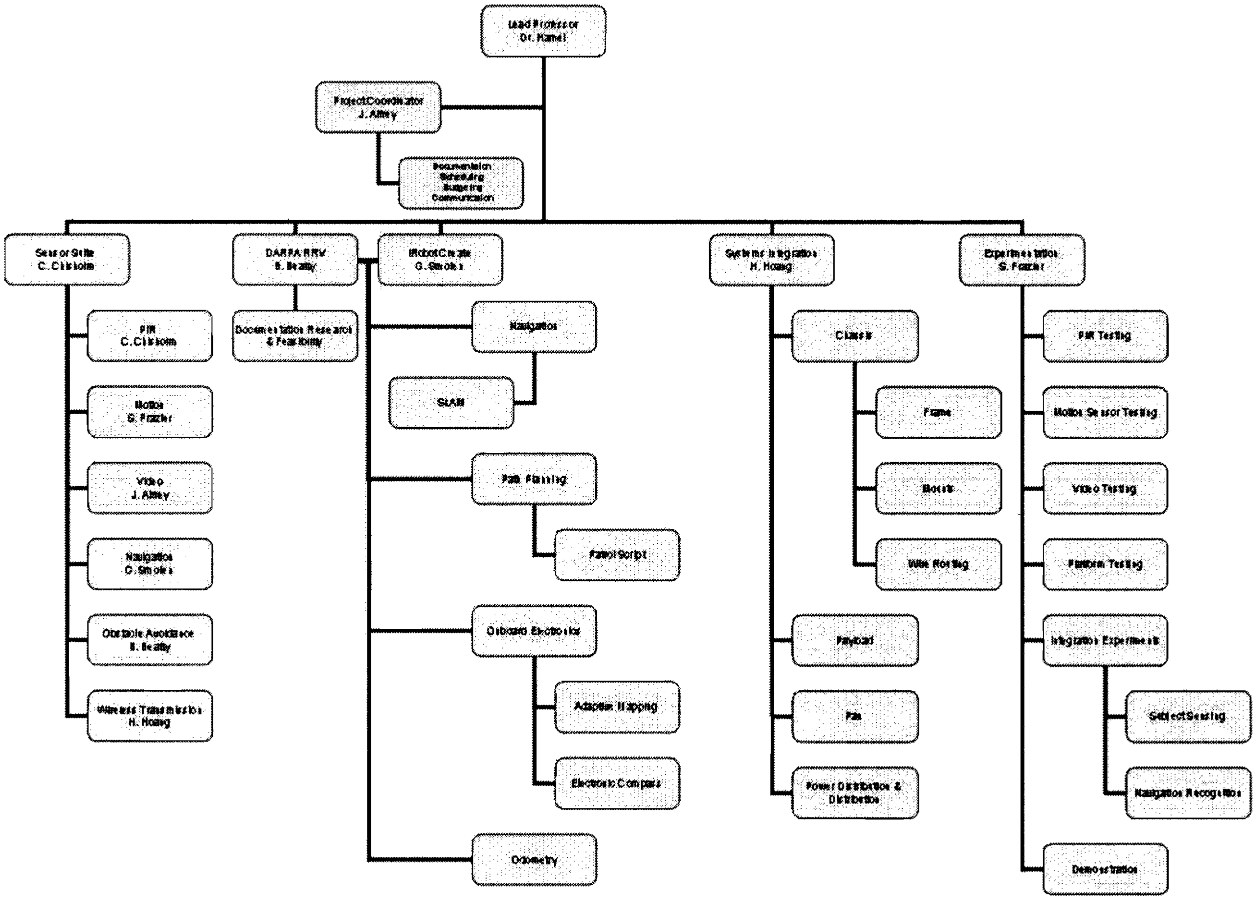
MoHSeR ACQUISITION LIST

ITEM NO	SYSTEM	DESCRIPTION	PART NO.	UNIT PRICE	NO.	TOTAL COST	STATUS
1	VIDEO	2.4GHz Wireless Night Camera	7545SC	109.99	1	109.99	To be ordered
2	VIDEO	2.4GHz Receiver	ZT-707	79.99	1	79.99	To be ordered
3	VIDEO	PCI	EDV-XV425	99.95	1	99.95	To be ordered
4	VIDEO	12V Battery Pack with charger and adapter	PR2310	89.95	1	89.95	To be ordered
5	PAN	SPG425 Servo 180 degree (w/+$30 for assembled system)*	HS-425BB	94.94	1	94.94	To be ordered
6	PAN	Battery & Charger*		50	1	50.00	TBD
7	PAN	Dual Servo Driver*	902MSD	39.99	1	39.99	TBD
8	INTEGRATION	USB to TTL Converter*	MCU-036-472	34.99	1	34.99	TBD
9	INTEGRATION	RS232 5V TTL 72in Type 1 Serial Converter Cable*	MCU-016-172	24.5	1	24.50	TBD
10	INTEGRATION	.118x24X96 Clear Lexan Sheet *		85.44	1	85.44	TBD
11	INTEGRATION	Other Misc Integration Expenses		100	1	100.00	TBD
12	INTEGRATION	Fabrication Estimate		150	1	150.00	TBD
13	SENSOR	MD3 Microwave Motion Sensor	MD3	49.99	1	49.99	To be ordered
14	INTEGRATION	OOPic-R Microcontroller	O-RS	84	1	84.00	TBD
15	NAVIGATION	Infrared beacon preprogrammed with byte number		7	10	70.00	TBD
16	NAVIGATION	Digital compass	CMPS03	60	1	60.00	TBD
17	OBJECT AVOIDANCE	Maxbotix LV-MaxSonar-EZ1 (sonic sensor)	LV-MaxSonar-EZ1	29.95	1	29.95	To be ordered

*Does not include S/H

Total: $1,253.68

7.4 Work Break-Down Structure



7.5 Project Management Techniques

Discussions among the design team have already suggested many schedule conflicts for the next phases. Currently, the only meeting time available for all six members occurs every Monday at 5:00 PM. Due to this limitation, it will require diligent communication on behalf of all of the team members and the ultimate responsibility for that relies on the project manager. In order to assist this effort, an online presence has been established using ZOHO – an online project management service. The site provides communication, calendar, task, meeting, and milestone functions in order to keep everyone in the project current and informed. The online presence is established and can be found at mohser.projects.zoho.com. This presence will allow a more efficient team structure. Each Monday will be the Project Review meeting with every member in attendance. As the name suggests, this will be the time to discuss, report, and address any open issues. The remainder of the week will allow for individuals or smaller teams to work on action items and projects. This work will all be traceable through ZOHO since it will be essential for everyone to report their work to this centralized location.

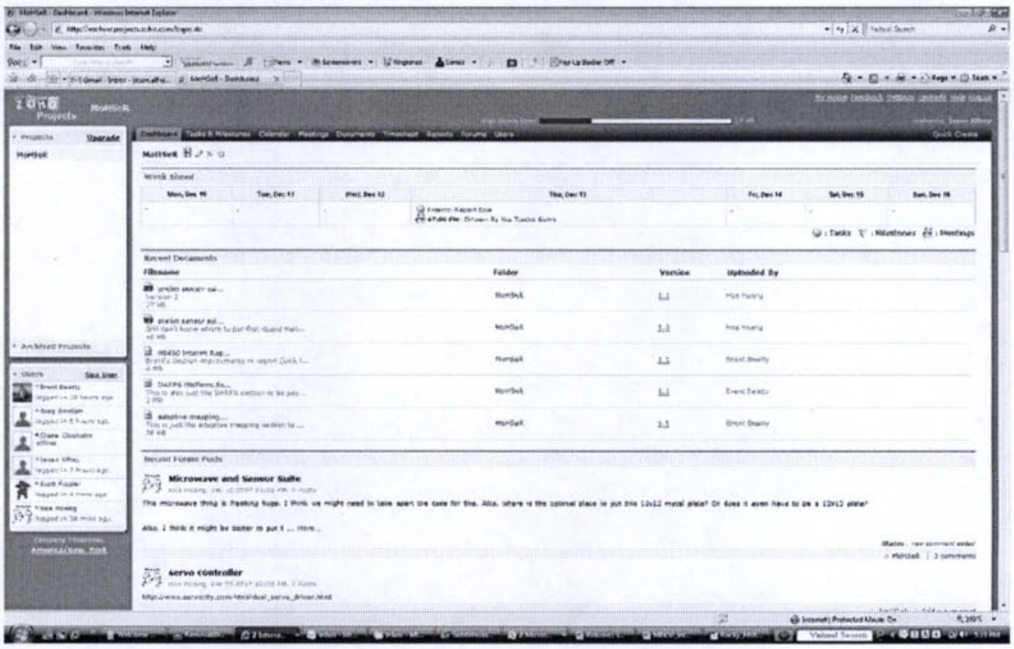


Figure 7.2 ZOHO provides a "Dashboard" to view upcoming milestones, tasks, meeting, as well as view recent forum discussions.

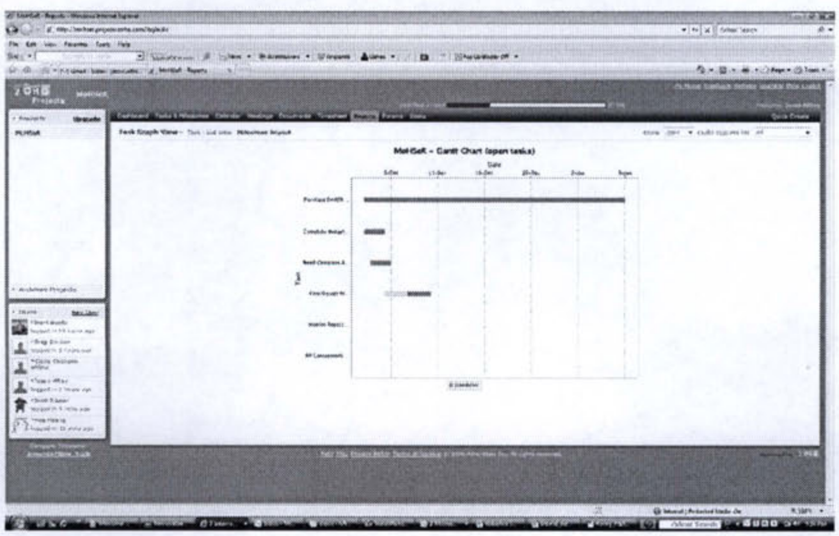


Figure 7.3: ZOHO provides Gantt chart capabilities.

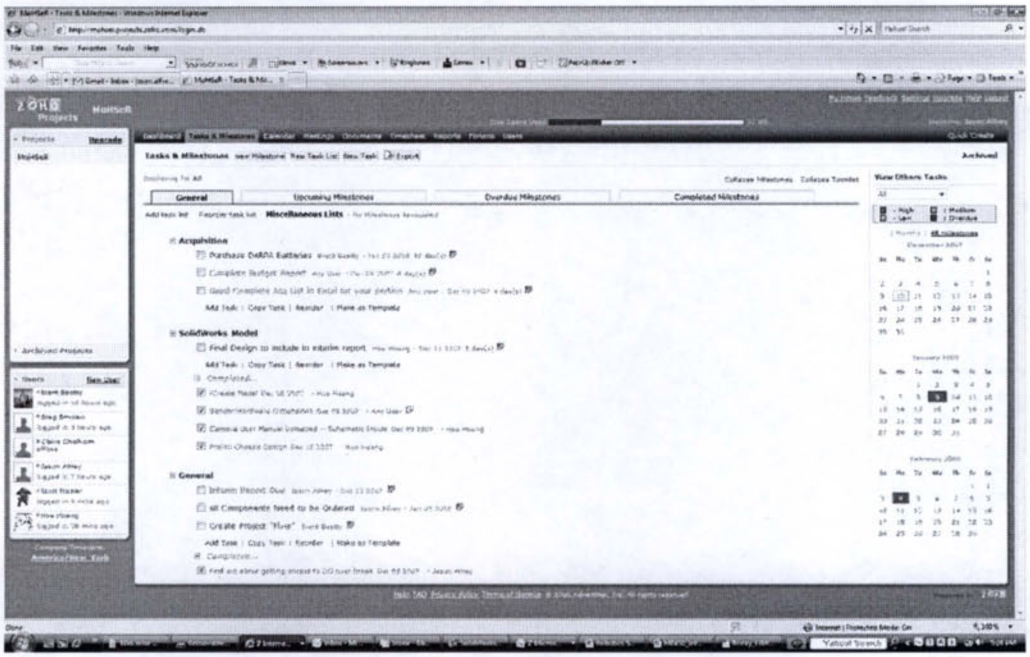


Figure 7.4: ZOHO makes it easy to assign and track Assigned Tasks

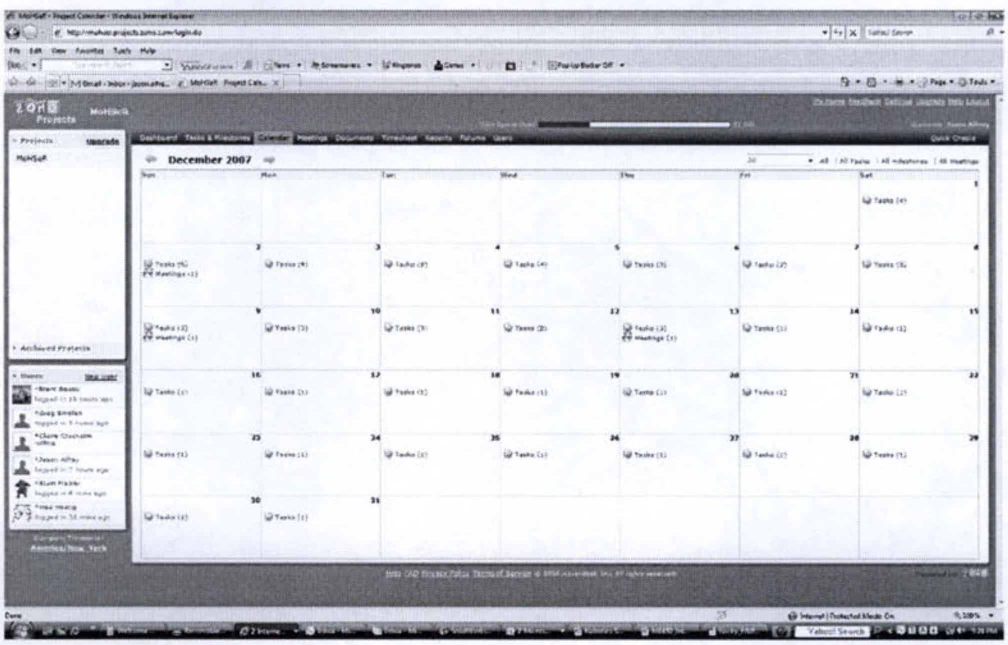


Figure 7.5: The Calendar allows one location for all members to schedule meeting times and locations, as well as view upcoming due tasks.

8. CONCLUSION

Although the MoHSeR design project is aggressive as has already been stated, the design team has continued to develop ideas on improving the project. For instance, it is envisioned that with time a more sophisticated form of mapping and localization be implemented. Likewise, the sensor suite was intentionally designed to be sizeable so that sensors can be integrated or removed as necessary. An example of a sensor that would be added would be a light detecting sensor. This design presumes that the heat signature, a human intruder, would be carrying a flashlight. There are a number of possibilities, but the primary focus of the design team is completing the next three phases of the design process successfully.

9. APPENDIX

A. REFERENCES

[1] M. Ashby, H. Shercliff, D. Cebon. *Materials: Engineering, Science, Processing and Design*. 2007, pp. 328-329.

[2] "Fresnel Lenses", Fresnel Technologies, Inc. 2003. http://www.fresneltech.com/pdf/FresnelLenses.pdf

[3] From a lecture given by Dr. Nofziger at the University of Arizona on Fresnel lenses. http://www.optics.arizona.edu/Nofziger/OPTI%20200/Lecture%2017/L17P2.htm

[4] http://acroname.com/robotics/parts/R3-PYRO1.html

B. CAMERA SPECIFICATIONS

Laguna Product No.: ZT-906T/Silver
Dimensions: 5" x 4" x 2 7/16"
Imaging Sensor: 1/4-inch CCD
CCD Total Pixels: NTSC: 512 (H) x 492 (V)
Horizontal Resolution: 420 TV lines
View Angle: 39°
Minimum Illumination: 1.0 Lux/F2.0 & 0 Lux (IR on)
Gain Control: Automatic
Channels: 4
Channel Frequency: Channel 1: 2,414 MHz; channel 2: 2,432 MHz; channel 3: 2,450 MHz;channel 4: 2,468 MHz
Transmission Power: 10 mW/CE; 2 mW/FCC
Modulation Type: FM
Bandwidth: 18 MHz
Power Supply: 12 V DC 500 mA
Consumption Current: 120 mA; 270 mA IR on
Unobstructed Effective Range: 100 minutes
LED Continuous Working Life: 6000 hours
Night Vision Range: 49 feet (15m)
Operating Temperature: -4° F to 122° F (-20° C to 50° C)
Relative Humidity: 20% to 80%
Approvals: CE, FCC approved

C. TRANSMITTER – ALTERNATIVE SOLUTION

XBee Radio Module Wireless Transmission (MaxStream)

The XBee Radio Module was another communication method that was considered. It uses the ISM 2.4 GHz like Bluetooth and Wireless LAN. The range is similar to Bluetooth at around 300 ft.

The XBee has been reported to be more reliable based on other users on the Create forum. Comparatively, XBee uses half the power that BAM would require. This is not like Wireless LAN because an XBee can only talk with another XBee. This requires an XBee Radio to be installed on the Create as well as a radio installed on the host PC. The host PC would require a TTL to RS-232 converter. The XBee requires 3V so resistors will be needed to limit the 5V from the Create. The XBee radio can be connected to the DB-25 connector in the cargo bay. There are, however, programs available that would relay the XBee to a network. The device connects to the TCP/IP port on the host computer that has an XBee Radio. The program relays anything from the connected clients to the XBee and anything from the XBee to any connected clients. While still requiring two XBee radios, this would allow any computer with a wireless internet connection to communicate and control the Create. The XBee can also be mounted on the OOPic (see OOPic section.) For radio connection, connect the DIN and DOUT on the XBee to the TX and RX pins on the OOPic. Two more pins will be needed to connect to the Create DB25 ports RX and TX pins. Lastly, supply power and ground and it is ready to operate. Programming can be written on the OOPic to talk to the Create and sensors and send to remote PC via radio.

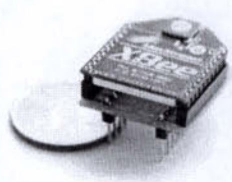
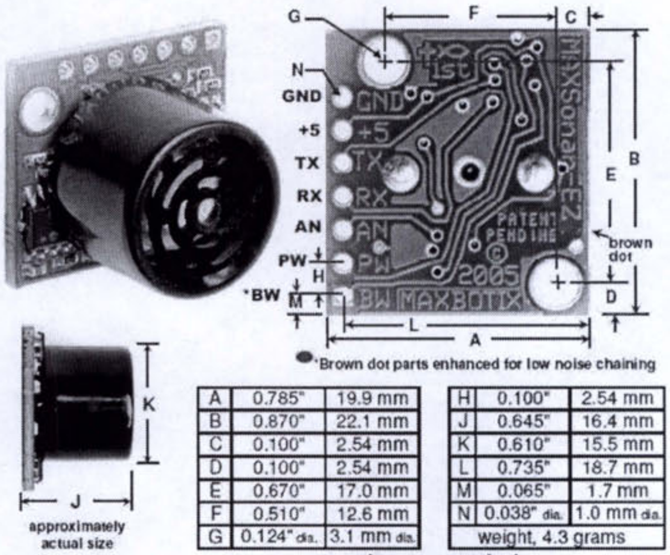


Figure C: XBee on breakout board to connect to OOPic

D. OBSTACLE DETECTION

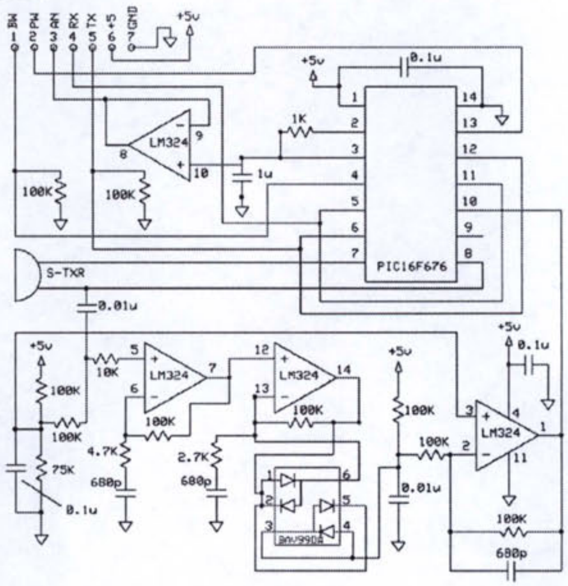
LV-MaxSonar-EZ1 Sensor



A	0.785"	19.9 mm	H	0.100"	2.54 mm
B	0.870"	22.1 mm	J	0.645"	16.4 mm
C	0.100"	2.54 mm	K	0.610"	15.5 mm
D	0.100"	2.54 mm	L	0.735"	18.7 mm
E	0.670"	17.0 mm	M	0.065"	1.7 mm
F	0.510"	12.6 mm	N	0.038" dia.	1.0 mm dia.
G	0.124" dia.	3.1 mm dia.	weight, 4.3 grams		

values are nominal

LV-MaxSonar-EZ1 Sensor Circuit:



E. BLUE TOOTH SPECIFICATIONS

LED Operation

The BAM's red LED indicates its current state:

- When the Create & BAM are first powered on, the LED will flash 26 times quickly.
- When the BAM is connected to a Bluetooth host, the LED will flash once per second.
- When the BAM is disconnected from a host, the LED flashes once every 3 seconds.

Bluetooth Settings

Device Name: Element Serial
Bluetooth PIN Code / Pass Code: 0000
Services Supported: SPP (Serial Port Profile)
RF Power: Class 1 Bluetooth

Host COM Port Settings

Baud: 57,600
Parity: None
Data Bits: 8
Stop Bits: 1
Hardware Flow Control: None
Software Flow Control: None

Electrical & Mechanical

Power Supply: 5.0 Vdc / 100mA (max)
Operating Temperature: 0 to 50° C
Storage Temperature: -20 to 60° C
Humidity: 0 to 80% RH
Internal Antenna Multilayer Chip, Peak Gain: 0.5dBi
Dimensions: 55x55x16 mm

F. NAVIGATION

Devantech Magnetic Compass

Voltage	5v
Current	20mA Typ.
Resolution	0.1 Degree
Accuracy	3-4 Degrees approx (after calibration)
Output 1	Timing Pulse 1mS to 37mS in 0.1mS increments
Output 2	I2C Interface, 0-255 and 0-3599
SCL Speed	up to 1MHz
Weight	0.03 oz.
Size	32mm x 35mm

Interfacing the compass to an OOPic (example taken from http://www.acroname.com/examples/10035/10035.html)

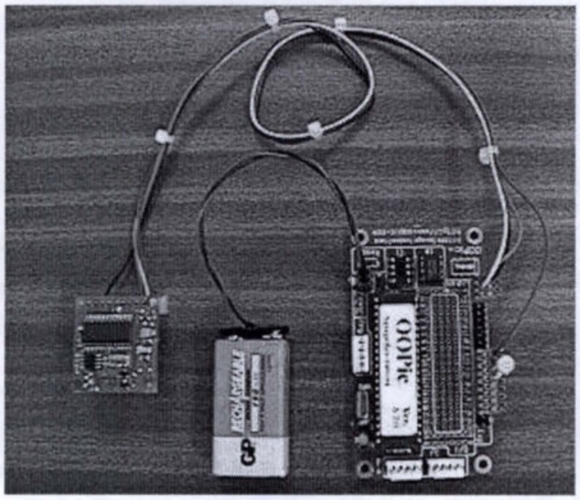
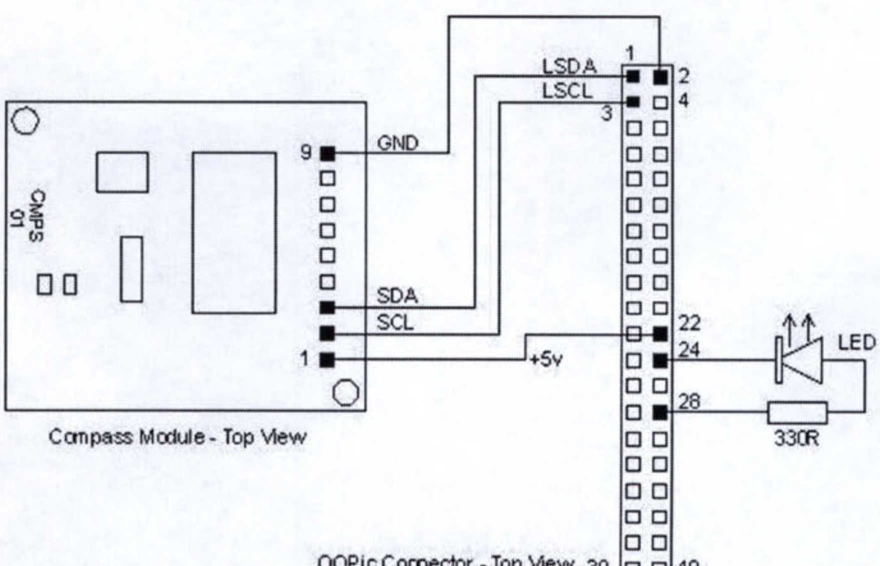


Figure F

In this example, an OOPic module uses a Devantech Compass Module to gain orientation information. The OOPic reads a 1-byte, 255-step compass reading representing one full revolution in headings from the compass module.
The compass module is connected to the OOPic via the OOPic's built-in I2C bus. You will also need a LED and a 330ohm resistor for this example. To make the connections, do the following steps in the order:

- Connect pin1 (5v) to pin22 (+5v) on the OOPic 40 way connector.
- Connect pin2 (SCL) to pin3 (LSCL) on the OOPic 40 way connector.
- Connect pin3 (SDA to pin1 (LSDA) on the OOPic 40 way connector.
- Connect pin9 (0v) to pin2 (GND) on the OOPic 40 way connector. Leave the other pins unconnected for now.
- Connect the LED cathode (shorter leg) to pin 24 on the OOPic.
- Connect the LED anode (longer leg) to the 330R resistor.
- Connect the other end of the 330R resistor to pin28 (I/O 30) on the OOPic.



IR Beacons

Parts:

PIC16F690
100ohm Resistor
IR LED
optional parts:
1K Potentiometer to adjust LED intensity
1K Resistor and a standard LED for Status LED

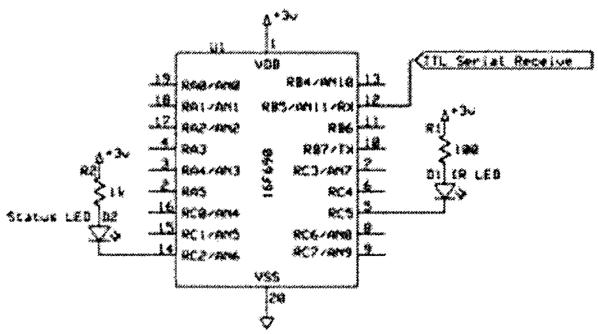


Figure G

Serial Protocol:
Send $FF for Command Mode
Send $01 to Enable+Set Beacon #
 Send 1-byte Beacon #
Send $02 to set Beacon Interval
 Send High Byte (milliseconds)
 Send Low Byte
Send $03 to disable Beacon so you can just use the serial port to send bytes without the Beacon interfering.

G. Source Code

OOPic example source code for PIR

This program pans a servo back and forth and takes readings from an Eltec 442-3 sensor mounted on the servo. The sensor responds to temperature changes in its field of view. A temperature change will cause the analog output of the sensor to produce a large peak (maximum) and a large dip (minimum). Otherwise, the output will stay around 2.5 volts. A large difference between the max and min indicates detection of something warm. The positions where the min and max occur give some indication of where the warm object is. Note

This example doesn't get into the hardware interface as it is pretty straightforward. The 3 pins on the BrainStem's Analog port provide power, ground, and the sensor output through direct connections to the sensor's 3 pins.

Contents

Source Code

```
/* pyro.tea */
/* pyroelectric scan test program */

#include <aCore.tea>
#include <aPrint.tea>
#include <aServo.tea>
#include <aA2D.tea>

#define PYROIN 0
#define SRVSCAN 3

#define SSTEP 1
#define SCANDELAY 30
#define DTHR 250

/* global variables for scan statistics */
int max;
int min;
int maxp;
int minp;
int pos1;
int pos2;
```

```
/* Resets scan statistics. */

void reset_scan()
{
 max = 0;
 min = 1023;
 maxp = 0;
 minp = 0;
}


/* This routine updates the servo position,
delays for a moment, then takes an A2D
reading from the Eltec sensor and updates
the global statistics variables. */

void update_max_min(int npos)
{
 int r;
 aServo_SetAbsolute(SRVSCAN, (unsigned char)npos);
 aCore_Sleep(SCANDELAY);
 r = aA2D_ReadInt(PYROIN);
 if (r > max) {
  max = r;
  maxp = npos;
 }
 if (r < min) {
  min = r;
  minp = npos;
 }
}


/* This routine checks the range between the peaks
and sees if it is big enough to be a valid detection.
If the detection is valid, the function returns the
average of the peak positions. */
int check_min_max()
{
 int pos = 0;
```

```
 int diff;
 diff = max - min;
 if (diff > DTHR) {
  pos = (maxp + minp) / 2;
 }
 return pos;
}


void main()
{
 int i;
 int pos1=0;
 int pos2=0;

 while (1) {
  // scan going in one direction
  reset_scan();
  for (i = 2; i < 254; i = i + SSTEP){
   update_max_min(i);
  }
  pos1=check_min_max();

  // scan going in the other direction
  reset_scan();
  for (i = 254; i >= 2; i = i - SSTEP) {
   update_max_min(i);
  }
  pos2=check_min_max();

  // print the position of the hot spot for each scan
  // (may vary a little bit depending on direction)
  aPrint_IntDec(pos1);
  aPrint_Char(',');
  aPrint_IntDec(pos2);
  aPrint_Char('\n');
 }
}
```